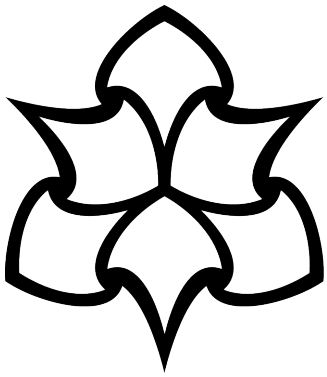


Novel Computerised Techniques for Recognition and Analysis of Diabetic Foot Ulcers

Manu Goyal

A thesis submitted in partial fulfillment of the requirements of
Manchester Metropolitan University
for the degree of Doctor of Philosophy

Department of Computing and Maths

MANCHESTER METROPOLITAN UNIVERSITY

2019

Abstract

Diabetic Foot Ulcers (DFU) that affect the lower extremities are a major complication of Diabetes Mellitus (DM). It has been estimated that patients with diabetes have a lifetime risk of 15% to 25% in developing DFU contributing up to 85% of the lower limb amputation due to failure to recognise and treat DFU properly. Current practice for DFU screening involves manual inspection of the foot by podiatrists and further medical tests such as vascular and blood tests are used to determine the presence of ischemia and infection in DFU. A comprehensive review of computerized techniques for recognition of DFU has been performed to identify the work done so far in this field. During this stage, it became clear that computerized analysis of DFU is relatively emerging field that is why related literature and research works are limited. There is also a lack of standardised public database of DFU and other wound-related pathologies.

We have received approximately 1500 DFU images through the ethical approval with Lancashire Teaching Hospitals. In this work, we standardised both DFU dataset and expert annotations to perform different computer vision tasks such as classification, segmentation and localization on popular deep learning frameworks. The main focus of this thesis is to develop automatic computer vision methods that can recognise the DFU of different stages and grades. Firstly, we used machine learning algorithms to classify the DFU patches against normal skin patches of the foot region to determine the possible misclassified cases of both classes. Secondly, we used fully convolutional networks for the segmentation of DFU and surrounding skin in full foot images with high specificity and sensitivity. Finally, we used robust and lightweight deep localisation methods in mobile devices to detect the DFU on foot images for remote monitoring. Despite receiving very good performance for the recognition of DFU, these algorithms were not able to detect pre-ulcer conditions and very subtle DFU.

Although recognition of DFU by computer vision algorithms is a valuable study, we performed the further analysis of DFU on foot images to determine factors that predict the risk of amputation such as the presence of infection and ischemia in DFU. The complete DFU diagnosis system with these computer vision algorithms have the potential to deliver a paradigm shift in diabetic foot care among diabetic patients, which represent a cost-effective, remote and convenient healthcare solution with more data and expert annotations.

Acknowledgements

Throughout my research and preparation of this thesis, many people have guided my understanding and knowledge that allowed me to form the work presented. Firstly, I would like to thank my Director of Studies, Dr. Moi Hoon Yap, for her kindness, patience and overall confidence in my abilities from the very beginning. I also want to express my utmost gratitude to my other supervisor Dr. Neil Reeves, and Dr. Satyan Rajbhandari who provided his timely perspective and help on the clinical aspects of my work and contributing greatly when I was in need.

I want to express special thanks to Lancashire Teaching Hospitals, and Jennifer Spragg for their extensive support and contribution in carrying out this research.

I have received help and advice, directly or indirectly, from my peers and colleagues throughout my study. These people motivated me to share ideas and build relationships to further expand my knowledge. From MMU, Ms. Jhan, Dr. Connah Kendrick, Mr. Sean Barton, Mr. Guido Ascenso, Dr. Adrain K. Davison, Dr. Brett Hewitt, Dr. Ezak Fadzrin, Mr. Nadim Baharum.

My most sincere thanks goes to my family, for always believing I could succeed, even when I did not. I am grateful for MMU for providing the studentship for me to complete this thesis, and provide valuable experience in teaching at a university level.

Contents

List of Figures

List of Tables

List of Abbreviations

2D 2-Dimensional

3D 3-Dimensional

AI Artificial Intelligence

AUC Area Under the ROC Curve

ANN Artificial Neural Networks

CML Conventional Machine Learning

CNN Convolutional Neural Network

CNNs Convolutional Neural Networks

DM Diabetes Mellitus

DFU Diabetic Foot Ulcers

FC Fully Connected

FCN Fully Connected Network

FCNs Fully Connected Networks

fps Frames per Second

FP False Positive

FPR False Positive Rate

FN False Negative

HOG Histogram of Oriented Gradients

ICT Information and Communication Technologies

IoT Internet of Things

IoU Intersection over Union

JSI Jaccard Similarity Index

LBP Local Binary Patterns

LRN Local Response Normalisation

MCC Matthew's Correlation Coefficient

ReLU Rectified Layer Unit

ROI Region of Interests

SD Standard Deviation

SGD Stochastic Gradient Descent

SIFT Scale Invariant Feature Transform

SMO Sequential Minimal Optimization

SURF Speeded Up Robust features

SVM Support Vector Machines

TP True Positive

TPR True Positive Rate

TML Traditional Machine Learning

TN True Negative

RF Random Forests

ROC Receiver Operating Characteristic

ROI Regions of Interest

XML Extensible Markup Language

List of Publications

This thesis is based on material from the following publications:

1. Manu Goyal, Neil D. Reeves, Adrian K. Davison, Satyan Rajbhandari, Moi Hoon Yap, "Robust Methods for Real-time Diabetic Foot Ulcer Detection and Localisation on Mobile Devices." *IEEE journal of biomedical and health informatics (2018).*

2. Manu Goyal, Neil D. Reeves, Adrian K. Davison, Satyan Rajbhandari, Jennifer Spragg, Moi Hoon Yap, "DFUNet: Convolutional Neural Networks for Diabetic Foot Ulcer Classification," *IEEE Transactions on Emerging Topics in Computational Intelligence (2018).*

3. Manu Goyal, Neil D. Reeves, Satyan Rajbhandari, Jennifer Spragg, Moi Hoon Yap, "Fully Convolutional Networks for Diabetic Foot Ulcer Segmentation," *IEEE International Conference on Systems, Man, and Cybernetics (IEEE SMC-2017).*

Dedicated to my Parents and Teachers

Chapter 1

Introduction

This Chapter outlines the background information of the project. The aim of this project is to implement computerised telemedicine systems which can detect DFU. This chapter starts with the introduction, motivation and problem statement of the DFU project. Then, contributions from each chapter and thesis organisation is discussed.

1.1 Background of DFU

DM commonly known as Diabetes is a lifelong condition resulting from hyperglycemia (high blood sugar levels), which leads to major life-threatening complications such as cardiovascular diseases, kidney failure, blindness and lower limb amputation which is often preceded by DFU [6]. According to the global report on diabetes in 2016 by the world health organisation, there were 422 million people suffering from DM in 2014, compared to 108 million people in 1980. Among the adults that are over 18 years of age, the global prevalence has gone up from 4.7% in 1980 to 8.5% in 2014 [7]. It is estimated by the end of 2035, the figure is expected to rise to 600 million people living with DM worldwide [8]. From this report, there is about only 20% of these people will be from developed countries and the rest will be from developing countries due to poor awareness and limited healthcare facilities [9]. There is about 15%-25% chance that a diabetic patient will eventually develop DFU and if proper care is not taken, that may result in lower limb amputation [10], although higher rates of up to 34% is suggested in the recent study [11]. Annually, on average, more than 1 million patients suffering

from diabetes lose part of their leg due to the failure to recognise and treat DFU appropriately [12]. A Diabetic patient with a 'high risk' foot needs periodic check-ups of doctors, continuous expensive medication, and hygienic personal care to avoid further consequences as discussed earlier. Hence, it causes a great financial burden on the patients and their family, especially in developing countries where the cost of treating this disease can be equivalent to 5.7 years of annual family income. Also, there is a large cost to healthcare systems in developed nations [13].

In current clinical practices, the evaluation of DFU comprises of various important tasks in early diagnosis, keeping track of development and number of lengthy actions taken in the treatment and management of DFU for each particular case: 1) the medical history of the patient is evaluated; 2) a wound or diabetic foot specialist examines the DFU thoroughly; 3) additional tests like CT scans, MRI, X-Ray may be useful to help develop a treatment plan. The patients with DFU generally have a problem of a swollen leg, although it can be itchy and painful depending on each case. Usually, the DFU have irregular structures and uncertain outer boundaries. The visual appearance of DFU and its surrounding skin depending upon the various stages i.e. redness, callus formation, blisters, significant tissues types like granulation, slough, bleeding, scaly skin. In the current healthcare settings, clinicians primarily monitor the patients by visual inspection to determine the important conditions such as area, depth, infection, ischemia, neuropathy, and site. There is a high risk of infection spreading in the body through DFU. Hence, patients need to visit the healthcare centres on regular interval for inspection of DFU which results in a financial burden to both patients and healthcare settings.

The proliferation of information and communication technologies present both challenges and opportunities in terms of the development of new age healthcare systems. Current literature of DFU evaluation with the help of computerised algorithms is still in the preliminary stage. Since the analysis of DFU with computerized methods is relatively emerging field, there are limited computer methods developed for the assessment of diabetic foot pathologies with the help of basic image processing and traditional machine learning [14, 15].

In recent years, there has been a rapid development in the area of computer vision, especially towards the difficult and important issues like understanding images of different domains such as spectral, non-medical objects, abnormalities

in medical imaging, and facial features recognition [16–19]. There is a major advancement of computer vision algorithms especially in the field of medical imaging. Recent advancement in deep learning has significantly improved the quality of these computer vision systems to detect the abnormalities in the different medical imaging such as Magnetic Resonance Imaging (MRI), dual-energy X-ray absorptiometry, ultrasonography, and computed tomography [20–24]. Although the potential for DFU analysis in computer vision is huge, core aspects of development need to be greatly improved to get accuracy rates of podiatrists. The major challenges in this field include the lack of publicly available datasets and expensive annotations. Hence, starting with end-to-end robust solutions for the recognition of different types of DFU on the substantial dataset would lay a foundation that would be beneficial to provide an initial point from where further interpretation can follow. Hence, developing the robust methods that can also be transferable to the mobile devices for the remote monitoring of DFU is an important advancement in computerised analysis of DFU.

But before we develop complete DFU diagnosis system to provide the outcome of DFU according to the different conditions such as area, depth, infection, ischemia, neuropathy, and site. There is a need for the robust methods with the help of cost-effective computer vision techniques to detect the DFU of various stages and grades according to the Texas classification [1, 25–27]. Since there are no automatic computerised solutions available so far in the literature survey which can analyse or detect the DFU on the basis of the medical classification system.

This thesis investigated fully automatic methods to detect DFU, with the potential to be applied in real heath-care settings. In the recent developments in computer vision and deep learning, it allowed us to design the end-to-end solutions for the recognition of DFU. We collected a large dataset of DFU of various patients of different backgrounds from Lancashire Teaching Hospital over a five years period. We received the NHS Research Ethics Committee approval with REC reference number 15/NW/0539 to use these images for our research. The ground truths for this DFU dataset were produced by the podiatrists expertise in DFU. The sample foot images in dataset are shown in the Fig. 1.1.

The main emphasis of this work was to clean the dataset and refine the expert annotations to perform three popular computer vision tasks for the medical imaging that are DFU classification, segmentation and localisation. Also, we converted

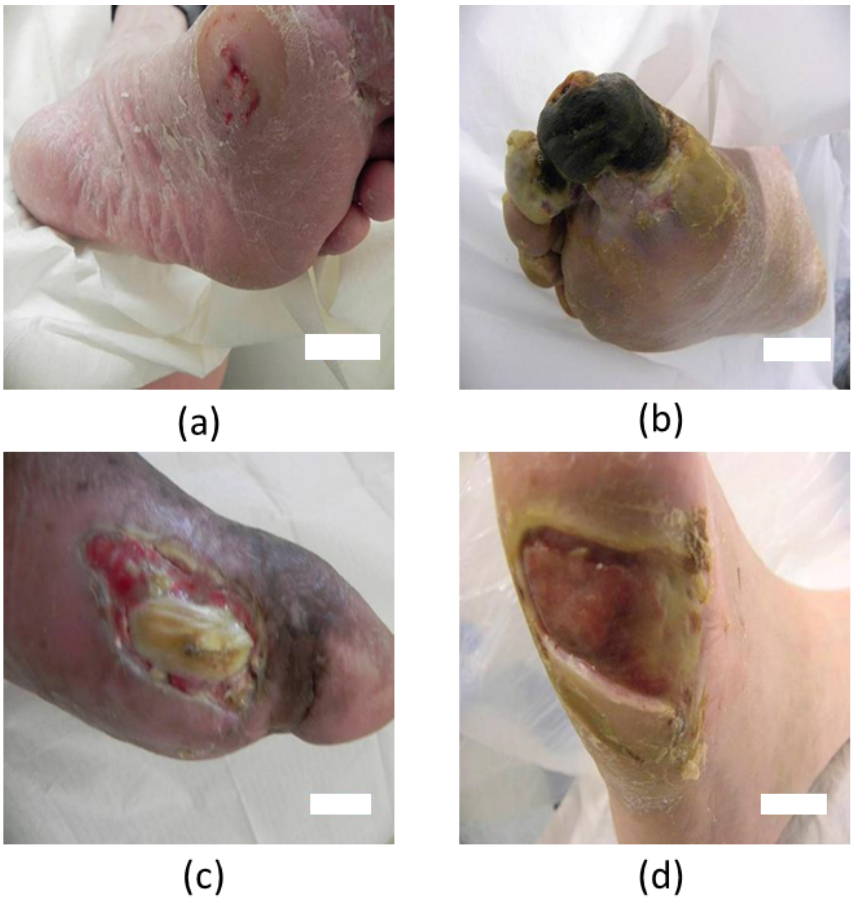


FIGURE 1.1: The sample images in the DFU dataset

these ground truths in popular annotation formats to support various deep learning libraries such as Caffe, PyTorch, Tensorflow. The major focus of this thesis was to design different types of end-to-end deep learning algorithms to achieve the recognition of DFU with high accuracy and precision. We compared the results achieved by these methods with current state-of-the-art methods (traditional machine learning and image processing). Another significant contribution was to transfer the robust DFU detection algorithms on mobile devices such as Nvidia Jetson TX2 and smart-phone application. These robust mobile applications could help patients and medical staffs to monitor the progress of DFU in the remote setting.

In the last contribution of this work, apart from just recognition of DFU in full foot images, we investigated the use of machine learning algorithms for the first time to determine ischemia and infection in DFU which could aid in predicting the outcome of DFU.

1.2 Motivation

The computerised methods mentioned in current literature based on traditional machine learning and image processing are not robust enough to detect the DFU of various grades and stages. Also, these methods are not end-to-end solutions as traditional machine learning algorithms usually consist of multiple stages such as pre-processing, feature extraction, training of classifiers, implementation of the classifier for recognition and post-processing whereas image processing techniques require manual tuning for individual images to get the final results.

The fast-growing research area of computer vision and medical imaging is largely driven by end-to-end algorithms with the potential of deploying these algorithms to real-world environments. The research based on recognition of DFU has focused on the ability of algorithms to detect DFU of varying grades and stages. Also, algorithms should be robust enough to detect the DFU of patients with different ethical backgrounds. Then, a further insight of DFU could be provided by determining the important conditions such as site, area, depth, infection and ischemia. Hence, designing robust computer vision algorithms that could analyse the DFU with high accuracy and precision have the potential to deliver a paradigm shift in diabetic foot care among diabetic patients, which represent a cost-effective, remote and convenient healthcare solution.

1.3 Problem Statement

The nature of an emerging field means that research is limited and tends to be exploratory rather than focused on already established work. In Chapter 2, current methods and algorithms for recognising, classifying and detecting DFU are discussed. Many different methods based on conventional machine learning and image processing are used to detect the DFU on the limited datasets. These literature works have not made their algorithms and datasets public. Hence, there is a need for end-to-end computerized solutions which can detect DFU of all grades and stages.

It is worth mentioning, there are no public DFU datasets available for research. Currently, most of the state-of-the-art computer recognition techniques

for medical imaging is based on deep learning models. Deep learning models imitate the functionality of the human brain to some extent with the help of neural networks. Hence, for training the deep learning models, there is a requirement of a large database of DFU images along with expert annotations. Expert annotations in medical imaging can be very expensive as there is a need for experienced clinicians to perform this to produce ground truth. In the current DFU dataset, these expert annotations are performed visually by podiatrists expertise in DFU. Furthermore, there can be influencing factors such as lighting conditions and skin tone due to the patient's ethnicity.

1.4 Aim and Objectives

The aim of this project was to develop novel computer algorithms to recognise and analyse DFU of various stages and grades. This thesis reviewed and critically analysed the existing computerised methods to identify DFU in terms of classification, object detection and segmentation. The following are the objectives:

1. To study the literature related to the background of DFU, medical classification systems for DFU, and computerised methods for recognition of DFU of various grades and stages.

2. To propose a novel computer vision method for DFU classification based on deep learning approach to differentiate normal skin lesions and DFU skin lesions in the foot region.

3. To develop new CNN-based automatic segmentation methods to segment DFU and surrounding skin on full foot images as surrounding skin is an important visual indicator to assess the progress of DFU.

4. To develop robust and lightweight deep learning methods for DFU localisation that can be utilized in mobile devices for remote monitoring.

5. To analyse the different conditions of diabetic foot pathologies according to the popular medical classification systems.

1.5 Thesis Contributions

The main contributions of this thesis are as follows:

1. We identified the research gaps in computerized methods for recognition of DFU, discussed various popular medical classification systems used to grade DFU and established standardised DFU datasets (with experts annotation) for popular computer vision tasks that are classification, segmentation and localisation.

2. The expert podiatrists delineated DFU dataset of 292 images to produce healthy skin and DFU skin patches. We used machine learning algorithms to extract the features for DFU and healthy skin patches to understand the differences in the computer vision perspective. A novel deep learning classification framework - DFUNet, which outperformed the state-of-the-art traditional machine learning and deep learning methods for DFU classification [2].

3. Experts precisely delineated the DFU and the surrounding skin region in full foot images. This is the first time, segmentation of surrounding skin is performed which is an important indicator for clinicians to assess the progress of DFU. We proposed to use two-tier transfer learning segmentation methods for semantic segmentation of DFU and its surrounding skin [3].

4. We used State-of-the-art deep learning localisation methods on the extensive DFU dataset of 1775 images and FootSnap dataset. We transferred the robust and lightweight models on mobile devices such as Nvidia Jetson TX2 and smart-phone android application for remote monitoring of DFU [1].

5. We investigated the different conditions of DFU such as site, infection, neuropathy, bacterial infection, area, and depth according to the computer vision perspective. In this work, we used machine learning algorithms to determine the important conditions of DFU such as bacterial infection and ischemia.

1.6 Thesis Organisation

This thesis is split into two main sections: introductory chapters and contribution chapters. The first section consists of three introductory chapters, the first of

which is the current Chapter introducing the work presented in the thesis and outlining what to expect from the research.

Chapter 2 presents fundamental knowledge and a review of the literature relating to DFU detection. Given the emergence of this field, some medical research is included to form a foundation on which DFU detection system should be based.

Chapter 3 provides technical information on the techniques explored for DFU detection. This includes techniques used for the image processing and traditional machine learning approaches, feature extraction methods and deep learning methods.

Chapter 4 provides details of DFU datasets that are used in the later contribution chapters. This also includes the format of expert annotations and performance metrics used for each DFU recognition tasks.

The second section includes four contribution chapters. Chapter 5 investigates the classification of DFU and healthy skin of the foot. The classification is completed using traditional machine learning, deep learning to classify these two classes. It also introduces our novel deep learning network called DFUNet which performed better than other deep learning and traditional machine learning methods.

Chapter 6 introduces an automated segmentation of DFU and its surrounding skin by using fully connected networks. We propose a two-tier transfer learning method by training the fully convolutional networks (FCNs) on larger datasets of images and use it as the pre-trained model for the segmentation of DFU and its surrounding skin.

Chapter 7 proposes the use of CNNs to localise DFU in real-time with two-tier transfer learning. To our best knowledge, this is the first time CNNs are used for this task. Since our main focus is on mobile devices, we emphasise on light-weight object localisation models. Finally, we demonstrate the application of our proposed methods on two types of mobile devices: Nvidia Jetson TX2 and an android mobile application.

The last of the contribution chapters, Chapter 8, investigate the use of machine learning algorithms to find the presence or absence of infection and ischemia in DFU, which are very important factors in determining the conditions of DFU

in medical classification systems such as Texas classification and Sinbad classification systems. We propose to use natural data-augmentation to avoid unnecessary artefacts in foot images and to have more balanced datasets. Then, we use both traditional machine learning and deep learning techniques to perform binary classification of ischemia and infection. In this experiment, the methods were able to perform better in the classification of ischemia and non-ischemia cases rather than infection and non-infection cases. We found that deep learning algorithms performed better for both classification tasks than traditional machine learning.

Finally, Chapter 9 concludes this thesis with a summary of contributions, the limitations faced in the field of DFU analysis and the future research direction.

Chapter 2

Clinical Background and Related Telemedicine Systems for DFU

An overview of the current literature related to popular medical classification systems to grade DFU, current telemedicine systems, and computerised methods for the recognition of DFU is presented.

2.1 Medical Classification Systems for DFU

The medical classification systems for DFU are used to classify the DFU on the basis of different conditions such as size, area, neuropathy, ischemia and infection to predict the outcome of DFU. These systems are all currently based on observations made by the clinician and clinical judgements. The popular medical classification systems are briefly explained below:

2.1.1 Wagner Classification System

The Wagner classification system is one of the most widely accepted classification systems which is based on the depth of penetration, the presence of osteomyelitis or gangrene, and the extent of tissue necrosis according to the following list [28, 29].

- Grade 0: No open lesions; may have deformity or cellulitis

- Grade 1: Superficial diabetic ulcer (partial or full thickness)
- Grade 2: Ulcer extension to ligament, tendon, joint capsule, or deep fascia without abscess or osteomyelitis
- Grade 3: Deep ulcer with abscess, osteomyelitis, or joint sepsis
- Grade 4: Gangrene localised to the portion of forefoot or heel
- Grade 5: Extensive gangrenous involvement of the entire foot

The main drawback of this classification system is that it does not address two important conditions that are ischemia and infection. This system classifies DFU on the basis of grades whereas Texas classification system provides the classification on the basis of both stages and grades [1, 29].

2.1.2 Texas Classification System

This standard classification system is popularly used by podiatrists and medical professionals to classify DFU into the different categories depending upon the stages and grades [1]. This system helps in evaluating the DFU according to the ulcer depth, the presence of infection and skin tissues types, and peripheral arterial occlusive disease in each category of the ulcer assessment. It consists of 4×4 matrix in which rows represent stages of the ulcer in alphabetic order and columns represent grades of the ulcer in numerical order. The stages of the ulcer are explained below:

- Stage A: No infection or ischemia
- Stage B: Infection present
- Stage C: Ischemia present
- Stage D: Infection and ischemia present

The different grades of the ulcer are:

- Grade 0: Epithelialised wound

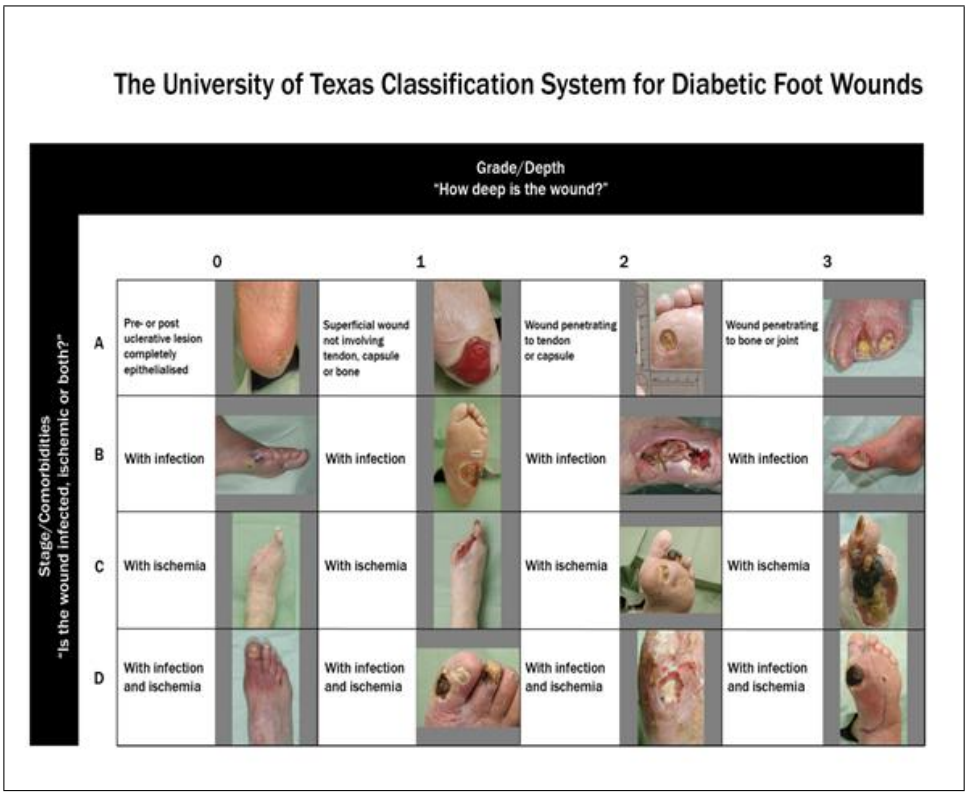


FIGURE 2.1: The University of Texas Classification System for DFU [1]

- Grade 1: Superficial wound
- Grade 2: Wound penetrates to tendon or capsule
- Grade 3: Wound penetrates to bone or joint

The complete Texas classification system for DFU is illustrated in Fig. 2.1

2.1.3 Sinbad Classification System

It is relatively new and simplified classification system introduced by Paul et al. [30] to compare the outcomes of DFU of different populations around the world. Sinbad score stands for S (Site), I (Ischemia), N (Neuropathy), B (Bacterial infection), A (Area), D (Depth). For each DFU, Sinbad score is calculated according to the Table 2.1. Although, Texas Classification System has been mostly used by podiatrists, but Sinbad scores are better suited for audit due to greater specificity [30]. Also for machine learning algorithms, the binary classification of each condition provided by this system is more suitable than other classification systems.

TABLE 2.1: The descriptions of SINBAD score according to the different conditions

Category	Definition	Sinbad score
Site	Forefoot	0
	Midfoot and hindfoot	1
Ischemia	Pedal blood flow intact: at least one pulse palpable	0
	Clinical evidence of reduced pedal blood flow	1
Neuropathy	Protective sensation intact	0
	Protective sensation lost	1
Bacterial infection	None	0
	Present	1
Area	Ulcer $\leq$ 1 cm	0
	Ulcer $>$ 1 cm	1
Depth	Ulcer confined to skin and subcutaneous tissue	0
	Ulcer reaching muscle, tendon or deeper	1
Total possible score		6

2.2 Telemedicine Systems

Telemedicine systems are the cost-effective healthcare services that are provided from the distance or remote location with the help of Information and Communication Technologies (ICT) [31]. With the recent development in ICT and limited healthcare services to the large population, the computerised telemedicine systems have great potential to overcome geographical distance barriers and provide cost-effective and quality healthcare services. Since there is a number of types of telemedicine systems suggested by various researchers and scientists over time. But, in general, the three main categories of telemedicine are store-and-forward, remote monitoring and real-time interactive services. Each of these telemedicine

systems has improved overall current healthcare systems and, it offers a number of benefits and facilities to medical staff and patients.

2.2.1 Store-and-Forward

The Store-and-forward telemedicine system is getting very popular as the applications of medical imaging are immensely improved. When utilised properly and with care, this practice can save time and cost of both medical practitioners and patients. Nowadays, the medical imaging modalities can also be recorded in electronic format and with history report and documentation, the patients don't need to meet impersonal with the medical specialist and practitioner every time [32]. Instead, the medical data such as biosignals or medical images of the patient can be sent to the specialist as needed with the help of communication devices in flash of a second. This telemedicine system is effective for various medical imaging modalities such as CT scan, X-Ray, MRI etc which is very common in the medical fields of dermatology, radiology and pathology [33–35].

2.2.2 Remote Monitoring

The remote monitoring telemedicine devices are very popular among patients. These devices allow patients to check the clinical signs or symptoms and monitor health without the need of any expert input [36, 37]. There is the number of self-monitoring kits available in the market to check the temperature of the body, sugar level, blood pressure, heart-rate which is effective in the management of various chronic diseases like diabetes, asthma, cardiovascular disease [38, 39].

There are the number of benefits of remote monitoring telemedicine systems which include cost-effective solutions, more frequent health check and greater patient satisfaction. But there is some risk of faulty telemedicine system or ineffective self-test conducted by patients can lead to inaccurate outcomes.

2.2.3 Real-Time Interactive Services

Interactive services can provide immediate advice to patients who require medical attention. There are several different mediums utilised for this purpose, including

phone, online and home visits. A medical history and consultation about presenting symptoms can be undertaken, followed by assessment similar to those usually conducted in face-to-face appointments. It also involves the automated solutions of clinical decisions to deal with the shortage of expert medical professionals in consultation for the various chronic diseases [40, 41].

These services are a great step forward in improving the accessibility of healthcare to all patients, particularly those living in areas with limited local healthcare settings. Additionally, these services offer a significant benefit of reduced cost in comparison to traditional in-person appointments.

2.3 Current Telemedicine Systems for DFU

This section focuses on the current telemedicine systems that are available for recognition and analysis of DFU. With the rapid growth in mobile telecommunications, remote communication is made possible with the help of standalone devices like smart-phones, laptops and the Internet. Nowadays, a pocket-size smart-phone with the advanced mobile operating system has the capability of a personal computer that can capture and send high-resolution pictures and also, audio and video communication with the help of advanced mobile internet like 4G. These telemedicine systems are broadly categorised into two categories:

- Non-automated Telemedicine Systems
- Automated Telemedicine Systems

2.3.1 Non-automated Telemedicine Systems

In the non-automated category, the common telemedicine systems based on these devices that are mostly set-up in the remote location for assessment of patients a) video conferencing [42]; b) 3-Dimensional (3D) wound imaging [43]; c) digital photography [44]; d) optical scanner [45]. However, there is a still need of specialised medical professionals on the other side for completing the assessment of the patient. In the recent study, Netten et al. [46] find that clinicians achieved low validity and reliability for remote assessment of DFU in foot images. Hence,

there is an urgent need for intelligent systems which can automatically detect the different DFU pathologies remotely.

2.3.2 Automated Telemedicine Systems

The use of automated telemedicine for DFU is still in its infancy. Notably, Liu et al. [14, 47] developed an intelligent telemedicine system for recognition of diabetic foot complications with the help of spectral imaging, infra-red thermal images and 3D surface reconstruction. However, to implement this system, there is a requirement for several expensive devices and specialist training to use these devices.

From a computer vision and medical imaging perspective, there are three common tasks can be performed for the recognition of abnormalities on medical images, which are 1) Classification 2) Localisation 3) Segmentation. These tasks on DFU are illustrated in Fig. 2.2. The current computer methods are based on manually engineered features or image processing approaches were implemented for tissue classification and segmentation of wound/ulcers. In general, virtually all the skin lesions related to both wound and ulcer are now termed as wound. In the medical perspective, both wound and ulcer are considered differently as wounds are caused by an external problem whereas, ulcers are caused by an internal problem. Also, there are differences in the appearance of the skin lesion of wound and ulcer, the cause (aetiology), the way the body responds (physiology) and disease processes (pathology) [48]. Hence, in this present study, only DFU are considered to determine how they are different from the normal skin at the same place of appearance.

The conventional machine learning for classification task was performed by extracting various features such as texture descriptors and color descriptors on small delineated patches of wound images, followed by machine learning algorithms to classify them into normal and abnormal skin patches [49–53]. As in many computer vision systems, the hand-crafted features are affected by lighting conditions and skin color depending upon the ethnicity group of the patient.

Various researchers have made contributions related to computerised methods for the recognition of DFU. We divided these contributions into the following four categories:

Classification Localisation Segmentation

FIGURE 2.2: Examples of three common tasks for abnormalities inspection on a DFU image. (a) Classification, (b) Localisation and (c) Segmentation of DFU (Green) and Surrounding Skin (Red) [25].

1. Algorithms development based on basic image processing and traditional machine learning techniques
2. Algorithms development based on deep learning techniques
3. Research based on different modalities of images
4. Smartphone applications for DFU

Algorithms development based on basic image processing and traditional machine learning techniques Several studies suggested computer vision methods based on basic image processing approaches and supervised traditional machine learning for the recognition of DFU/wound. Mainly, these studies have performed the segmentation task by extracting texture descriptors and color descriptors on small patches of wound/DFU images, followed by traditional machine learning algorithms to classify them into normal and abnormal skin patches [50–53]. In conventional machine learning, the hand-crafted features are usually affected by skin shades, illumination, and image resolution. Also, these techniques struggled to segment the irregular contour of the ulcers or wounds. On the other hand, the unsupervised approaches rely on image processing techniques, edge detection, morphological operations and clustering algorithms using different color space to segment the wounds from images [54–56]. Wang et al. [15] used an image capture box to capture image data and determined the area of DFU using cascaded two-stage SVM-based classification. They proposed the use of superpixel technique for segmentation and extracted the number of features to perform two-stage classification. Although this system reported promising results, it has not been validated on a more substantial dataset. In addition, the image capture box

is very impractical for data collection as there is a need for the patient's barefoot to be placed directly in contact with the screen of image capture box. In healthcare, such a setting would not be allowed due to the concerns regarding infection control.

The majority of these methods involve manually tuning of the parameters according to different input images and multi-stage processing which make them hard to implement in clinical settings. These state-of-the-art methods were validated on relatively small datasets, ranging from 10 to 172 images. Current state-of-the-art methods based on basic image processing and traditional machine learning techniques are not robust, due to their nature of reliance on specific regulators and rules, with certain assumptions.

Algorithms development based on deep learning techniques In contrast to traditional machine learning, deep learning methods do not require such intense assumptions and have demonstrated superiority in object localisation and segmentation of DFU, which suggests that the robust fully automated recognition of DFU may be achieved, by adopting such approach [25, 26, 57]. In the field of deep learning, several researchers made contributions to the classification and segmentation of DFU. Goyal et al. [26] proposed a new deep learning framework called DFUNet which classified the skin lesions of the foot region into two classes, i.e. normal skin (healthy skin) and abnormal skin (DFU). In addition, they used deep learning methods for the semantic segmentation of DFU and its surrounding skin with a limited dataset of 600 images [25]. In one of the recent works [58], the deep localisation networks are designed to detect the DFU with great accuracy and these algorithms are transferred to the mobile systems such as smart-phone and Nvidia Jetson TX2 to assist remote monitoring. Wang et al. [57] proposed a new deep learning architecture based on encoder-decoder to perform wound segmentation and analysis to measure the healing progress of the wound. To date, this work is the first attempt to develop deep learning methods for the DFU localisation task.

Research based on different modalities of images Then, in a separate study from computer vision techniques, Van et al. [59] proposed the recognition of DFU using a different modality called infra-red thermal imaging. They found that there is a significant temperature difference between the DFU and the surrounding

healthy skin of the foot. Hence, they used this considerable temperature difference on a heat-map to detect the DFU. Liu et al. presented a preliminary case study to evaluate the effectiveness of infra-red dermal thermography on diabetic feet soles to identify pre-signs of ulceration [60]. Harding et al. [61] performed a study to assess the infra-red imaging for the prevention of secondary osteomyelitis. Similarly, infra-red thermography has been used in various studies to detect the complications related to the DFU [62, 63].

Smartphone applications for DFU Health applications on the smartphone are fast becoming popular in monitoring essential aspects of the human body. Yap et al. [64, 65] developed an app called FootSnap, which is used to produce the standardised dataset of the DFU images. This application used basic image processing techniques such as edge detection to provide the ghost images of the foot which is useful to monitor the progress of DFU. Since this was designed to standardising image capture conditions, it did not perform any automated DFU recognition. Recently, Brown et al. [66] developed a smartphone application called MyFootCare, which provides useful guidance to the DFU patients as well as keep the record of foot images. In this application, the end-users need to crop the patch of the captured image, and with basic color clustering algorithms, it can produce DFU segmentation. But, previous research [25] has already shown that the basic clustering algorithms are not robust enough to provide accurate DFU segmentation on full foot images.

2.4 Research Direction

With limited healthcare settings and increasing global population and financial burden, the medical facilities to the patients are becoming big concerns for even the developed countries. The computerised telemedicine systems are often tipped as potential solutions to this problem. The proliferation of Information and Communication Technologies (ICT) present both challenges and opportunities in terms of the development of new age healthcare systems.

For any computerised DFU recognition algorithm, there could be many challenges that are needed to resolve before ensuring an effective recognition system with the help of DFU images. Based on a review of the current literature, a few

research challenges are identified that are (1) high inter-class similarity between the normal (healthy skin) and abnormal classes (DFU) in the foot region; (2) intra-class variations depending upon the classification of DFU; (3) lighting conditions; (4)patient's ethnicity [26, 27]. Similarly, the changes in visual appearance of DFU and its surrounding skin i.e. redness, callus formation, blisters, significant tissues types like granulation, slough, bleeding, scaly skin depending upon the various stages remains another challenge for robust recognition of DFU. Hence, the DFU analysis and recognition with the help of computer vision algorithms could be very challenging tasks. Producing ground truths for the segmentation of DFU and its surrounding skin which usually have very irregular structures and uncertain outer boundaries that makes it very challenging annotation task for podiatrists.

Regardless of current issues, DFU recognition systems are still in the early stages of development, with the recognition algorithms are tested on very small testing sets. Hence, computerised algorithms need to train and test on substantial DFU datasets to provide robust recognition of DFU. Further, due to the medical data and copyright, the current literature did not make their dataset public.

But before we move straight forward to predict the outcomes of DFU according to the medical classification systems, there is a need of the robust methods with the help of cost-effective computer vision techniques to detect the DFU of various stages and grades according to the Texas classification [1, 25–27]. These robust methods could help for clinical applications which can help medical staff to monitor the real-time progress of DFU in the remote setting. Since there are no technical or computerised solution available so far in the literature survey which can analyse the DFU on the basis of the medical classification system. Analysing DFU with the help of computer vision has the potential to deliver a paradigm shift in diabetic foot care among diabetic patients, which represent a cost-effective, remote, and convenient healthcare solution.

2.5 Summary

This Chapter investigated the DFU background in both medical and computer algorithms perspective and why it is important to develop the intelligent telemedicine systems for recognition and analysing the DFU. It started with a discussion of current medical practice for the evaluation of DFU of patients on the basis of different

medical classification systems such as Wagner, Texas, and Sinbad. According to these systems, DFU can be classified into many categories depending on the different conditions such as neuropathy, ischemia, size, area, depth, infection. Then, we outlined the challenges faced by the computerised systems for recognition and analysis of DFU.

In the next section of this chapter, the current literature based on non-automated and automated telemedicine system for DFU is discussed. In automated category, the methods based on basic image processing and machine learning techniques make many assumptions in the recognition of DFU, and in contrast, whereas deep learning algorithms are more suitable approach for DFU recognition. With the rapid growth in computer vision techniques for the recognition of abnormalities in medical imaging, we identified key factors for the recognition of DFU. So far, there are no technical solutions to detect the outcome of DFU according to the medical classification systems. Recognition and analysis of DFU with images could be difficult even for medical staff, but approaches to detect DFU using computer vision are steadily growing, but have a long way to go before being as well-established as analysis according to the medical classification systems.

Chapter 3

Theories and Techniques

This Chapter focuses on the theories and techniques used throughout the thesis, including basic image processing, traditional machine learning in terms of feature descriptors and classifier, deep learning methods that are used for the computer vision tasks for the recognition of DFU.

3.1 Image Processing and Traditional Machine Learning

Basic image processing includes analysing and manipulating digital images with the help of computer systems and algorithms. These algorithms perform some operations on the input image in order to extract some useful information or to get an enhanced image. There are many useful applications of image processing in medical imaging which mainly focuses on improving the quality of input images such as contrast enhancement and distinguish the Regions of Interest (ROI) in an image as computer vision tasks [67–71].

Machine learning is an application of artificial intelligence algorithms that provides computer systems with the ability to automatically learn and improve from more data without being explicitly programmed [72, 73]. Machine learning algorithms are usually divided into two categories that are supervised learning and unsupervised learning as shown in Fig. 3.1.

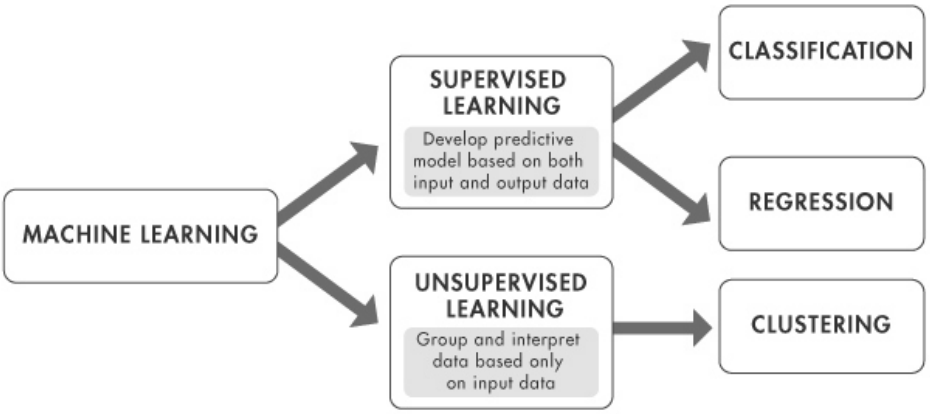


FIGURE 3.1: Classification of Machine Learning algorithms

Supervised machine learning algorithms train on the labelled data to predict future events. Supervised learning algorithms are used for different types of data such as text, audio, images and videos [74–76]. But, in our work, we used this learning for the images, and videos. Also, supervised learning is used in both traditional machine learning and deep learning used for DFU recognition. It starts with annotated training dataset, the learning algorithms extract features from the training data to build an inferred function to make predictions about the output values. At the end of each iteration, the systems compare its output with the ground truths (intended output) and use error to improve the model accordingly. After sufficient training and minimal errors, the algorithms can provide outputs for any unseen input data. Supervised learning can be further classified into classification and regression techniques to develop predictive algorithms. Hence, machine learning algorithms use supervised learning on the images data to learn the features or pattern of certain objects and then use it to perform inference on unseen data based on the previous examples that we provide as shown in Fig. 3.2.

Classification techniques are used to classify the data into categories, for example, whether a foot image has a presence of DFU or not or whether dermoscopic image containing mole is cancerous or benign [23, 77–79]. These are the examples of binary classification problem, it can be further divided into more categories such as medical classification system such as the Wagner system classifies DFU into 6 categories whereas the Texas classification model uses 16 categories [1, 28].

Classification models predict discrete responses in terms of categories, regression techniques predict continuous responses in a certain range, for example, predicting the age of human using face images [80]. The continuous output of

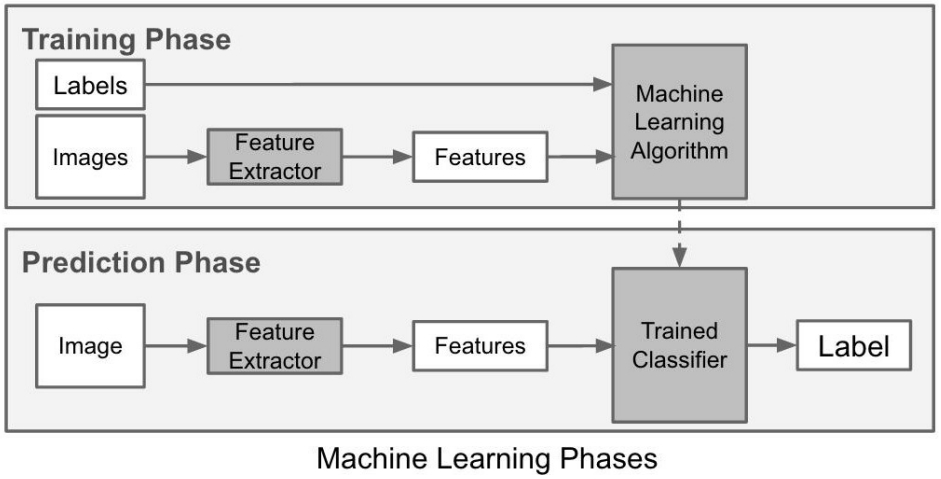


FIGURE 3.2: Training and inference using supervised machine learning algorithm

regression models is a real-value, such as an integer or floating point value. These are often quantities, such as amounts and sizes.

Unsupervised learning algorithms find hidden patterns or intrinsic structures in non-labelled data to predict the output. Hence, these algorithms are used to draw inferences from datasets consisting of input data without labelled responses. Clustering is the most common unsupervised learning technique. It is used for exploratory data analysis to find hidden patterns or groupings in data. Applications for clustering include gene sequence analysis, market research, and object recognition [81, 82].

3.1.1 K-Mean Clustering

In image processing methods, segmentation is very important to segment the region of interest in medical imaging with the help of clustering algorithms [83]. K-means clustering algorithm is one of the important technique used in image segmentation. K-mean clustering is an unsupervised clustering algorithm which is based on inherent distances between data points to classify these points into multiple clusters. K-mean algorithm is an iterative method which calculates the new cluster centres in each phase and reassigns every pixel to their nearest cluster centre [84, 85]. Many state-of-the-art algorithms used K-mean clustering algorithm to segment the DFU from foot images as shown in Fig. 3.3.

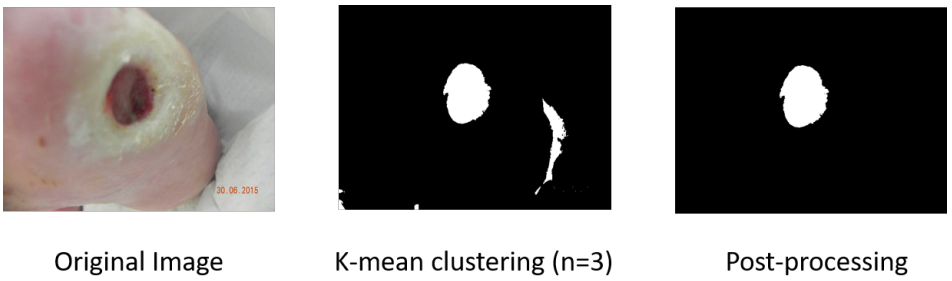


FIGURE 3.3: DFU recognition using K-mean clustering (n=3) and post-processing

3.1.2 Feature Descriptors

A feature descriptor is an algorithm which takes an image as input and outputs feature vectors. Feature descriptors find hidden patterns in the training images and convert them into a series of numbers that act as a sort of numerical "fingerprint" that can be used to differentiate one feature from another. Feature descriptors are instrumental for many computer vision tasks such as image classification, registration, object detection, object tracking, 3-D construction. Over the years, the researchers have introduced number of feature descriptors descriptors such as edge detection, corner detection [86], texture descriptors such as Local Binary Patterns (LBP) [87], Gabor filter [88], Histogram of Oriented Gradients (HOG) [89], shape-based descriptors such as Hough transform [90] and color descriptors such as Normalised *RGB, HSV, and L*u*v* features [91] to perform these tasks. These feature descriptors transform the input image data into a set of features known as the feature vector. Feature vectors are the higher level representation of data in the given dataset. In this thesis, we used feature descriptors for DFU classification task that consists of two classes as healthy skin and DFU skin. Since there are mainly textural and color differences due to the changes of the visual appearance of healthy skin to DFU skin depending upon the significant tissues such as callus formation, blisters, granulation, slough, bleeding, scaly skin. Hence, we focused on the color descriptors and texture features as feature descriptors for DFU classification. The Local Binary Patterns (LBP) [92, 93] and Histogram of Oriented Gradients (HOG) [94] which encodes information about the local neighbourhood image gradients are the commonly used texture features for the classification task. For the color descriptors, we used color histograms and the mean values of each channel in various color spaces such as RGB, HSV, L*u*v as feature vectors for this classification.

3.1.3 Local Binary Patterns

The Local Binary Patterns (LBP) [92] operator forms labels for each pixel in an image by thresholding a 3×3 neighbourhood of each pixel with the centre value. A result is a binary number where if the outside pixels are equal to or greater than the centre pixel, it is assigned a 1, otherwise, it is assigned a 0. The amount of labels will ,therefore, be $2^8 = 256$ labels.

This operator was extended to use neighbourhoods of different sizes. Using a circular neighbourhood and bilinearly interpolating values at non-integer pixel coordinates allow any radius and number of pixels in the neighbourhood. The grey-scale variance of the local neighbourhood can be used as the complementary contrast method. The following notation of (P, R) will be used for pixel neighbourhoods, where P are sampling points on a circle of radius R [92, 93, 95].

Uniform patterns are used to reduce the length of the overall feature vector and implement a single rotation-invariant descriptor. A LBP that is uniform when the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. So 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010010 (6 transitions) are not. In the computation of the LBP labels, uniform patterns are used so that there is a separate label for each uniform pattern and all the non-uniform patterns are labelled with a single label. For example, when using $(8, R)$ neighbourhood, there are a total of 256 patterns, 58 of which are uniform, which yields in 59 different labels [92, 96, 97].

Each region has the standard LBP operator applied with c being the centre pixel and P being neighbouring pixels with a radius of R

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p \tag{3.1}$$

where g_c is the grey value of the centre pixel and g_p is the grey value of the p-th neighbouring pixel around R. 2^p defines weights to neighbouring pixel locations and is used to obtain the decimal value. The sign function to determine what

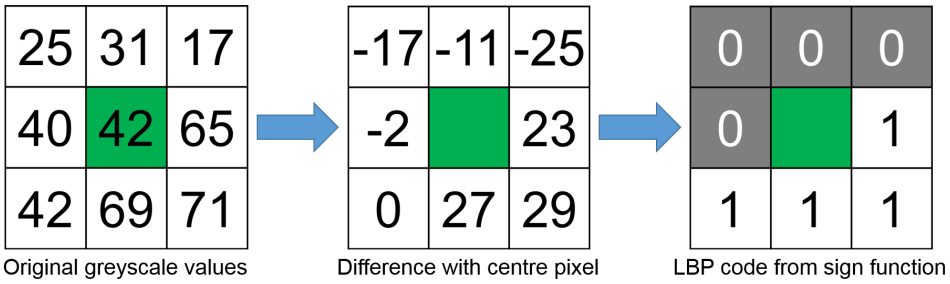


FIGURE 3.4: LBP code calculation by using the difference of the neighbourhood pixels around the centre.

binary value is assigned to the pattern is calculated as

$$s(\mathbf{A}) = \begin{cases} 1, & \text{if } \mathbf{A} \geq 0 \\ 0, & \text{if } \mathbf{A} < 0 \end{cases} \tag{3.2}$$

If the grey value of P is larger than or equal to c, then the binary value is 1, otherwise it will be 0. Fig. 3.4 illustrates the sign function on a neighbourhood of pixels. After the image has been assigned LBPs, the histogram can be calculated by

$$H_i = \sum_{x,y} I\{LBP_l(x,y) = i\}, i = 0, \ldots, n-1 \tag{3.3}$$

We used the LBP feature descriptor to find the feature vectors for the two healthy skin patches and one DFU patch from foot region. Then, the squared error of LBP histograms is compared between normal patches and normal vs ulcer as shown in Fig. 3.5. In Fig. 3.5, we found out the squared error of LBP histograms is very high in normal vs ulcer compared to the normal vs normal. Hence, LBP is utilized as one of the feature descriptors in the DFU classification.

3.1.4 Histogram of Oriented Gradients

Histogram of Oriented Gradients (HOG) features [94] were originally created for human detection in 2-Dimensional (2D) images and used the pixel orientation values, weighted by its magnitude, to calculate features for describing a human as an object. Fig. 3.6 shows a visualisation of a normal skin image with the HOG operator applied whereas Fig. 3.7 shows HOG visualization of ulcer skin patch in the foot region. The image shown is for understanding how the features are applied and are not used in processing. The small white plots on the image denote the

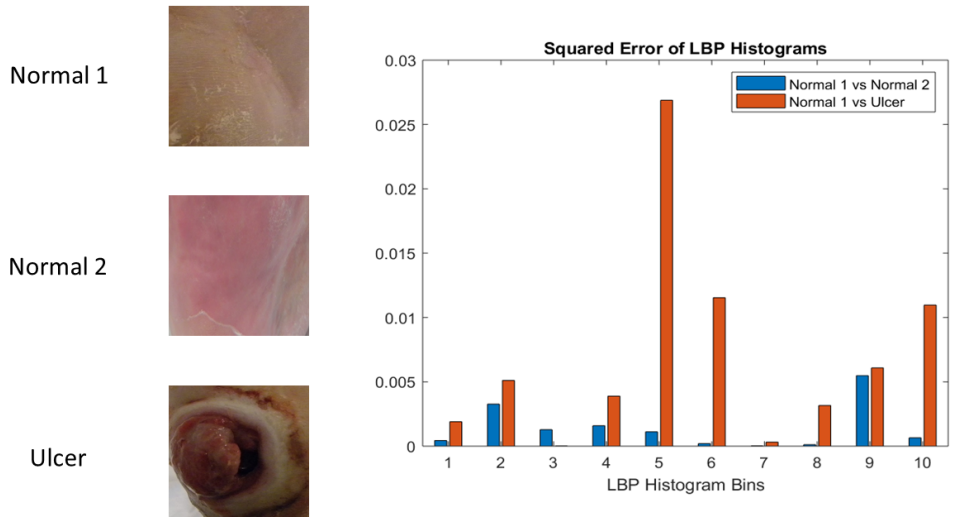


FIGURE 3.5: LBP code calculation by using the difference of the neighbourhood pixels around the centre.

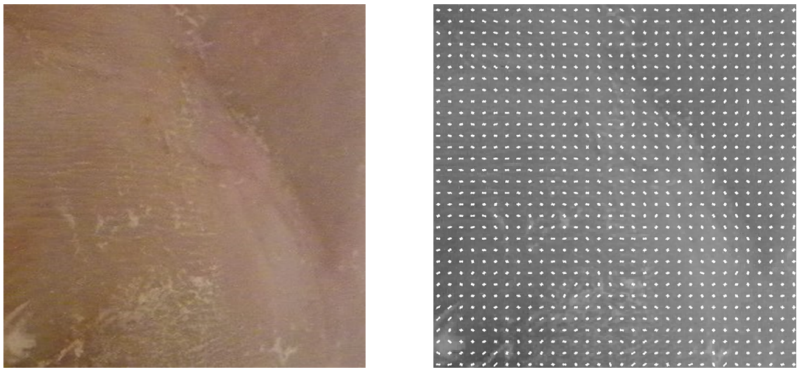


FIGURE 3.6: HOG visualization on normal skin patch.

direction of a HOG cell weighted by the pixel magnitude using signed calculations. So, the longer the white line, the higher the magnitude in that direction. Each white line represents a particular bin, and in this example, there are 9 bins in a 360 degree (or 2π) available orientations split into 40 degrees per bin. As shown in the DFU skin patch in Fig. 3.7 and healthy skin in Fig. 3.6, there is a difference in the pixel orientation values especially around the contour of DFU and its surrounding skin.

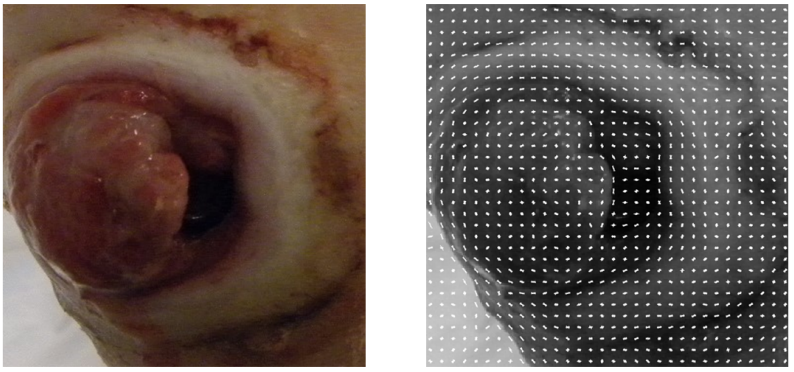


FIGURE 3.7: HOG visualization on abnormal skin DFU patch.

3.1.5 Color Descriptors

The color descriptors are proved to be effective feature descriptors in the classification tasks where there is a significant difference in color between the classes. DFU develops over the healthy skin of the foot with different tissues formation depending on the various grades and stages of DFU. These tissues have significant color differences when compared to healthy skin. The three color space that we have used: *RGB*, *HSV* and *L*u*v* in which we computed color histograms, as well as a most dominated color value, in each channel are utilized as feature vectors for the identification of DFU.

3.1.6 Support Vector Machines

Machine learning algorithms are all about learning structure (pattern and information) from raw data, and many methods exist in this field. In this section, the classification method, Support Vector Machines (SVM), is described. SVM can be utilized for both classification and regression tasks. For DFU classification, we utilized SVM as a classifier to train feature vectors extracted by feature descriptors mentioned in the previous sections.

First proposed by Cortes and Vapnik [98] a SVM attempts to find a linear decision surface (hyperplane) that can separate classes and has the largest distance between support vectors (elements in data closest to each other across classes). If

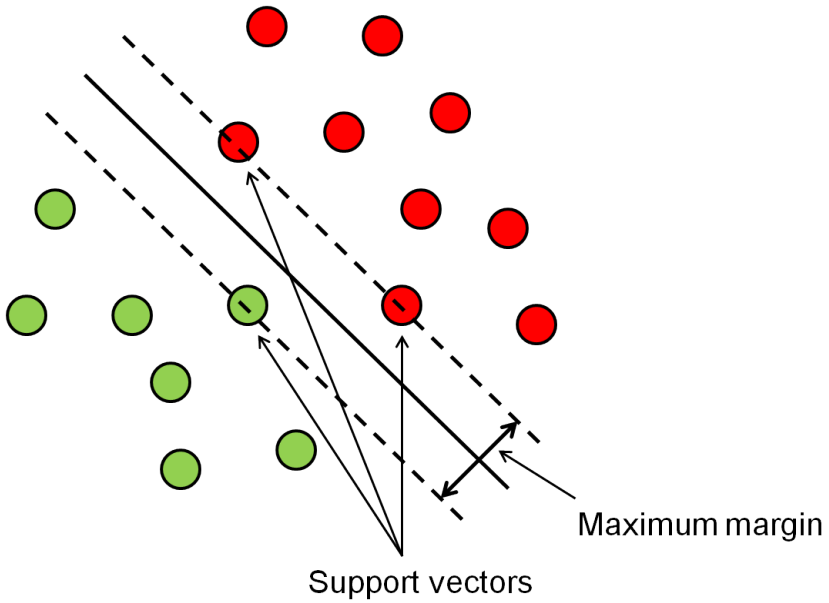


FIGURE 3.8: Visualisation of an SVM hyperplane. The green and red circles represent the positive and negative classes respectively, with the support vectors contributing to hyperplane separation leading to the determination of the maximum margin [99].

a linear surface does not exist, then a SVM is able to use kernel functions to map the data into a higher dimensional space where a decision surface can be found. SVM was originally based on the Structural Risk Minimisation principle, which was used for machine learning from a finite dataset.

As shown in Fig. 3.8, data points are split using an optimal separating hyperplane. The dashed lines on either side of the hyperplane is hereby defined as the margin m. Each training vector (feature vector) $\mathbf{x}$ belongs to a class y, with the training set defined as $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$. The total set and classes are defined as $(\mathbf{x}_i) \in \mathbb{R}^d$ and $y_i \in \{-1, +1\}$ where $\mathbb{R}^d$ is a real number in d-dimensions and $\{-1, +1\}$ are the two classes. For a given hyperplane, $\mathbf{x}_+$ and $\mathbf{x}_-$ are the closest points to the hyperplane among the positive and negative examples. The norm of a vector $\mathbf{w}$ is denoted by $||\mathbf{w}||$ as its length and is given by $\sqrt{\mathbf{w}^T\mathbf{w}}$. A unit vector $\mathbf{w}$ in the direction of $\mathbf{w}$ is given by $\mathbf{w}/||\mathbf{w}||$ and $||\mathbf{w}|| = 1$.

From a geometric consideration, the margin of a hyperplane h with respect to a dataset D can be defined as

$$m_D(f) = \frac{1}{2}\mathbf{w}^T(\mathbf{w}_+ - \mathbf{w}_-) \tag{3.4}$$

where there is an assumptions that $\mathbf{w}_+$ and $\mathbf{w}_-$) are equidistant from the decision boundary as

$$f(\mathbf{x}_+) = \mathbf{w}^T\mathbf{x}_+ + b = a \tag{3.5}$$

$$f(\mathbf{x}_-) = \mathbf{w}^T\mathbf{x}_- + b = -a \tag{3.6}$$

for some constant $a > 0$. To make this geometric margin meaningful, the value of the decision for the points closest to the hyperplane, $a = 1$. By adding Eq. 3.5 and Eq. 3.6 and then dividing by $||\mathbf{w}||$, the margin becomes

$$m_D(f) = \frac{1}{2}\mathbf{w}^T(\mathbf{w}_+ - \mathbf{w}_-) = \frac{1}{||\mathbf{w}||} \tag{3.7}$$

Next, a maximum margin classifier, sometimes called a hard margin, is defined to handle linearly separable data. It can then be modified to attempt to handle less easily separable (or non-separable) data. The maximum margin classifier is the discriminant function that maximises the geometric margin $1/||\mathbf{w}||$ which is the equivalent to minimising $||\mathbf{w}^2||$. This leads to the following constrained optimization problem

$$\begin{aligned} \min_{\mathbf{x},b} \quad & ||\mathbf{w}^2|| \\ \text{subject to} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, i = 1, 2, \ldots, n \end{aligned} \tag{3.8}$$

where the constraints show ensure that the maximum margin classifies each example correctly assuming the data is linearly separable. However, it is often the case that data is not linearly separable. A larger margin can be determined by allowing for some misclassification of points. The optimization problem now becomes

$$\begin{aligned} \min_{\mathbf{x},b} \quad & \frac{1}{2}||\mathbf{w}^2|| + C\sum_{i=1}^{n}\xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, \xi_i \geq 0 \end{aligned} \tag{3.9}$$

where $\xi \geq 0$ are the variables that allow for a margin error, $0 \leq \xi_i \leq 1$, or to be misclassified by $\xi > 1$. The constant $C > 0$ sets the relative importance of maximising the margin and minimising the amount of errors. This way of calculating for non-separable data is called a soft margin SVM.

Lagrange multipliers are used as a mathematical method to solve constrained optimization problems of differentiable functions. With an SVM, the saddle point

of the Lagrange function can be found using

$$L(\mathbf{w}, b, \alpha) = ||\mathbf{w}^2|| - \sum_{i=1}^{n} \alpha_i \{y_i[(\mathbf{w}^T \cdot \mathbf{x}_i) + b] - 1\} \tag{3.10}$$

where α_i are the Lagrange multipliers. The Lagrangian function has to be minimised with respect to $\mathbf{w}, b$ and maximised with respect to $\alpha_i \geq 0$. The optimization can be transformed into its dual problem as

$$\begin{aligned} \max_{\alpha} \quad & L(\mathbf{w}, b, \alpha) = \max_{\alpha} \sum_{i=1}^{n} \alpha_i - \sum_{i,j=1}^{n} \alpha_i \alpha_j Y_i Y_j K_{ij} \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C \,\& \quad \sum_{i=1}^{n} \alpha_i Y_i = 0 \end{aligned} \tag{3.11}$$

and the optimal separating hyperplane is represented by the dual solution

$$\mathbf{w} = \sum_{i=1}^{n} \alpha_i \cdot y_i \cdot \mathbf{x}_i \tag{3.12}$$

The value of b can be estimated by inputting $\mathbf{w}$ into the original equation $\mathbf{w}^T\mathbf{x} + b = 0$. For testing, the classification is given by

$$f(\mathbf{x}) = sign(\mathbf{w} \cdot \mathbf{x} + b) \tag{3.13}$$

for any new data point $\mathbf{x}$. If the training data input into the SVM is non-separable, then the error variables, ξ, can be used.

3.2 Convolutional Neural Networks

Artificial Intelligence (AI) is a new buzzword as an emerging technology in recent years. Many giant multi-national companies such as Amazon, Facebook, Google are investing a large number of resources in this technology as many expert tips AI technology to improve human life in almost every evitable sector. With all commercial and scientific fields announcing all the advances that AI technology is making, still, we are scratching the silver surface of the huge potential of AI technology. Also, there is a common fear among the people, it would make the human workers obsolete in the future. But in reality, AI technology is most likely

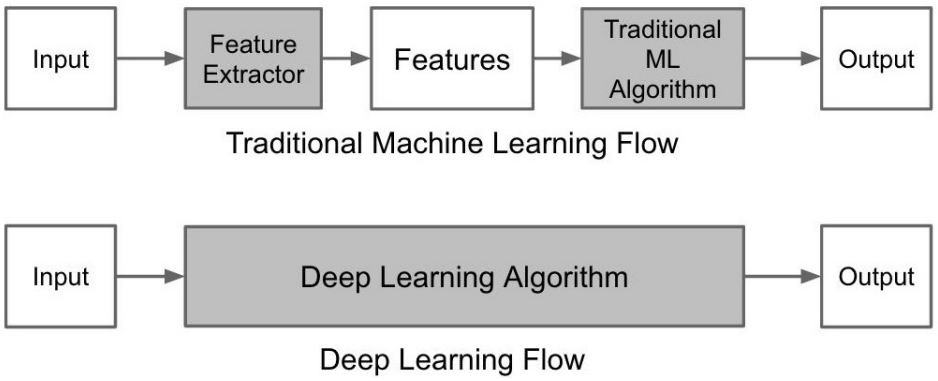


FIGURE 3.9: Supervised machine learning and deep learning algorithm

to assist the human force to improve their work and lifestyle in coming future intervention.

The AI technology relies on the number of vast assisting technologies and algorithms such as robotics, machine learning, Internet to perform the tasks. It imitates the working of the human brain to perform the tasks in a similar way the human does. But with the backing of unlimited computational power and storage capacity, these technologies have the potential to outperform the human. Deep learning algorithms are the latest revelation in AI technology mainly to perform computer vision and speech recognition with the help of camera and microphone respectively. In speech recognition, the devices can now understand many world languages and also can provide you with useful information. With the help of computer vision algorithms, the computer is able to see and learn about the common things around us. It basically, uses similar feature representation such as color, shape, pattern to recognise the different objects [100]. In traditional machine learning, the features and classifiers are manually selected by the users to train the model whereas in deep learning techniques act as bit of black box which extracts features by its own with the help of convolutional layers and in later part, fully connected layers with Softmax classifier to predict the outputs as shown in the Fig. 3.9.

Below are the few examples of AI projects that use deep learning to identify specific objects include:

1. Technology to enable self-driving cars [101].
2. Speech recognition algorithms used to interact with humans such as SIRI, Alexa, and Google Assistant [102].

3. Assisting medical experts for the recognition of abnormalities in medical imaging.

4. Identification of 120 breeds of dog; one algorithm has been reported to have achieved an accuracy of more than 96% [103].

5. Prediction of user choices and demands such as Netflix predicts the movies list that user may like to watch in future and similarly, Uber predicts the period of high demand for taxis [104].

The CNNs are neural networks which are based on a computational model inspired by the working of biological brains. A large number of connected nodes called artificial neurons are used in these systems similar to biological neurons in the brain. These systems are based on the supervised learning in which they learn the features from the manual labelled data such as "dog" or "no dog" without any prior knowledge about the dog. It uses these learned features to provide inference on unseen examples. Similarly, in medical imaging, these networks are used to determine "abnormal" or "normal" in various modalities of imaging. They are used to train on large labelled databases of different medical images and matched or exceeded the expert vision for the recognition of objects in the images [105, 106].

1. Breast cancer

2. Brain tumour

3. Skin cancer

4. Alzheimer disease

These algorithms will soon be scalable to multiple devices, platforms, and operating systems, reducing their cost and increasing their availability for diagnosis and research. Universities, governments and research funding agencies have recognised the opportunities to improve early diagnosis of cancer, heart disease, diabetes and dementia among others and are investing heavily in the sector. AI techniques approved by the US Food and Drug Administration (FDA) for clinical use by September 2018 include products to:

1. Identify signs of diabetic retinopathy in retinal images [107].

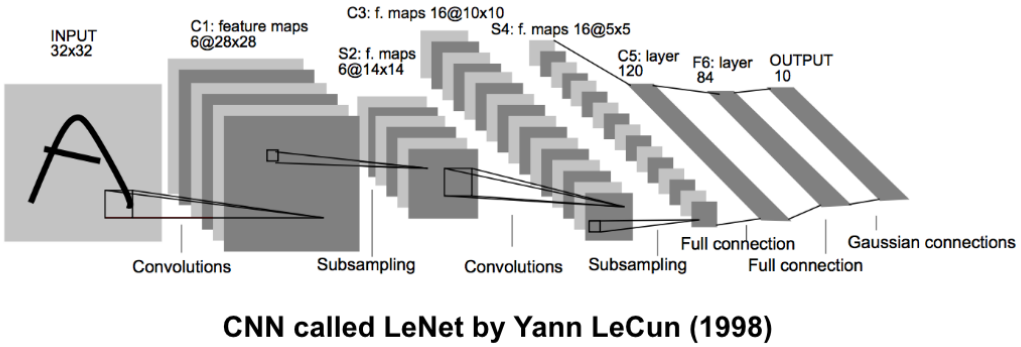


FIGURE 3.10: The overview of convolutional neural network LeNet designed by LeCun [2]

2. Recognise signs of stroke in CT scans [108].

3. Visualise blood flow in the heart [109].

4. Skin vision mobile app uses AI to detect skin cancer [110].

3.2.1 Introduction and Background

The very first CNN was designed by Yann LeCun for the text recognition in 1998 [2]. The overall overview of this CNN is demonstrated in Fig. 3.10. The CNN really provide the breakthrough when AlexNet emerged as the winner of imageNet ILSVRC-2012 competition in classification category [3]. In recent years, CNNs are developed to perform the number of tasks other than classification such as regression, segmentation, object detection and localization, object landmarking, object tracking, object activity recognition. In this thesis, we used CNNs for three categories that are classification, segmentation, and localization.

CNNs are like Artificial Neural Networks (ANN), both make use of neurons that have learnable weights and biases. The main difference between these networks is input data. Unlike ANN, where the input data is 2-D vector, here for CNNs, the input data is generally multi-channelled image data (RGB images in our case). Hence, input image data is firstly down-sampled to 2-D feature vector with intermediate layers such as convolutional layer and pooling layer. CNNs are usually composed of four types of layers that are convolutional layer, pooling layer, ReLU layer, and fully connected layers. These layers are stacked on each other in different settings to design different CNN architectures.

3.2.1.1 Convolutional Layer

Convolutional layers extract the feature map from the images. Convolution preserves the spatial relationship between pixels by learning image features using small squares of input data. Initial convolutional layers extract low-level features like edges, corners and shapes. With later convolutional layers, high-level features can be interpreted by the network to detect the type of class.

A convolution is a kernel-based method to detect features in an image, it uses a small sliding matrix (kernel) over a matrix (input image) and compute element-wise multiplication between these two matrices, and add the element-wise multiplication outputs to get the final output which forms the final output. In a CNN, a convolution layer has adjustable parameters, such as:

- *Kernel Size*: This value determines the kernels height and width in pixels. Adjusting the filter size influences the networks ability to determine specific features in an image. In CNN, 3×3 filter is one of the most commonly used kernel size for the convolutional layer.
- *Stride*: This value determines how much the kernel slides after each calculation. By increasing the stride, input data to convolutional layer can also be down-sampled to form smaller feature maps as it begins to jump pixels at a time.
- *Padding*: Padding allows the input data to have additional zeros placed around the edge, to maintain the similar feature map size despite using a larger kernel for convolution.
- *Filter Count*: It determines the number of kernels is used in the convolutional layer. By using more array of kernels, a convolutional layer can be used to identify a multitude of different features as each kernel can focus on highlighting alternative feature.

An example of visualization of the convolutional layer on a DFU image is shown in Fig. 3.11.

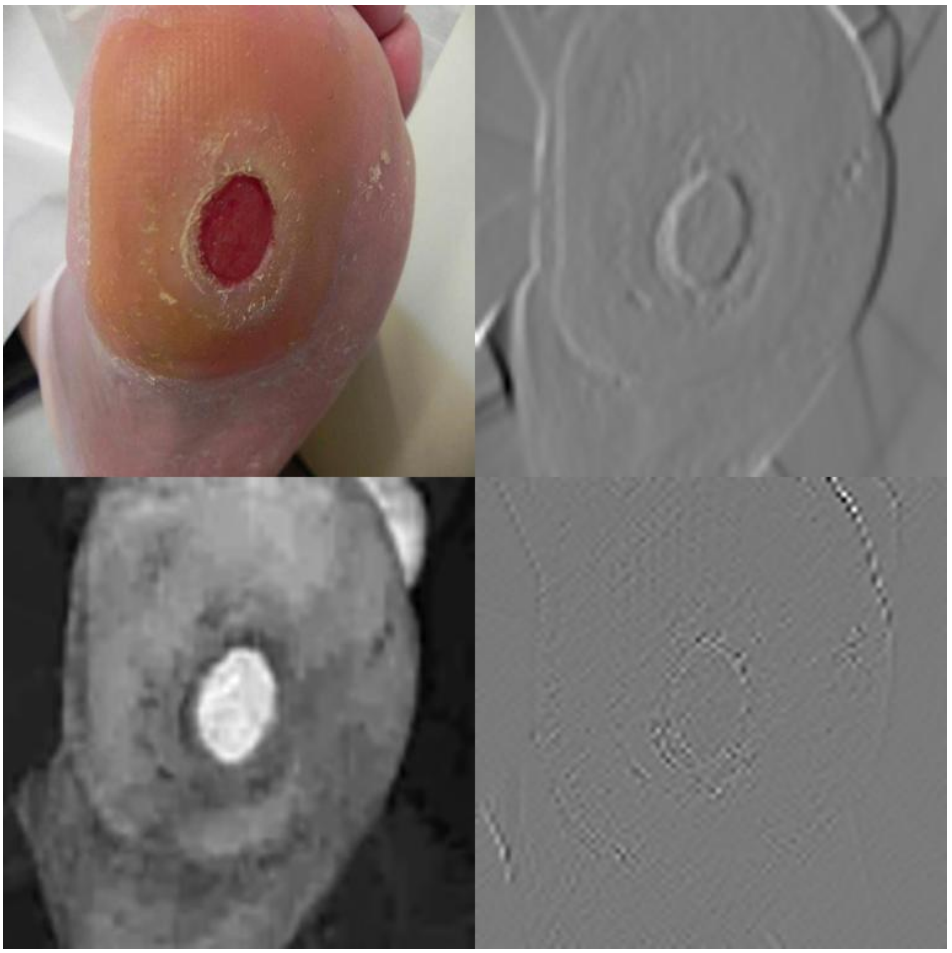


FIGURE 3.11: The visualization of some feature outputs of Ist convolutional layer of AlexNet on sample DFU image [3]

3.2.1.2 Activation Functions

An activation function is used after each convolutional layer to determine the values of the feature map such as replacing all the negative values in the feature map by zero. The activation functions can be divided into two categories that are: (1) Linear activation functions; (2) Non-linear activation functions. The output of the linear function will not be confined between any range (-infinity to infinity). The non-linear functions are the most used activation functions as it helps the model to generalize even the difficult data. A commonly used non-linear activation function is sigmoid, illustrated in Eq. 3.14. The sigmoid function normalises the data between the range of 0 and 1 similar to the probability.

The ReLU is most commonly used activation function in the CNNs as described in Eq. 3.15. The ReLU is half rectified (from bottom) i.e. f(x) is zero when x is less than zero and f(x) is equal to x when x is above or equal to zero.

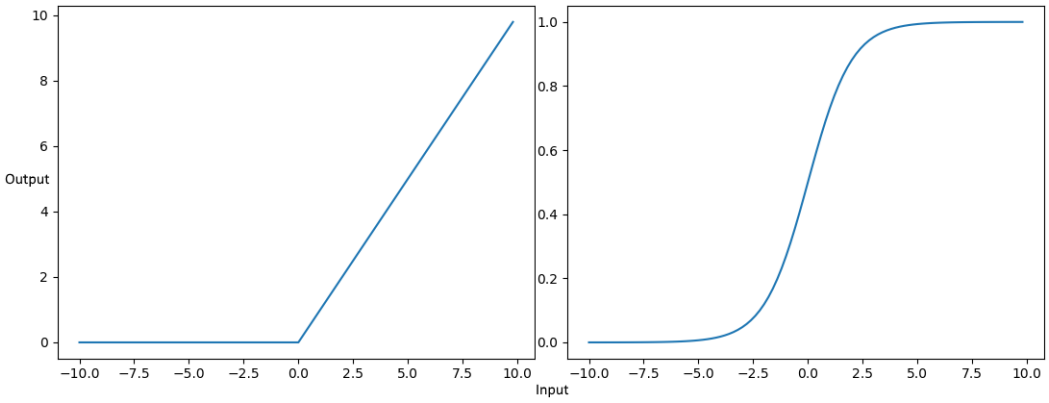


FIGURE 3.12: This image shows ReLU (left) activation vs sigmoid (right), notice how sigmoid normalises the range, but ReLU allows an output range between 0 and infinity

FIGURE 3.13: The example of activation of last ReLU layer of AlexNet on sample DFU image

The activations of the ReLU layer after final convolutional layer in AlexNet clearly pinpoint areas of the DFU image that have strong features as shown in Fig. 3.13.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{3.14}$$

$$f(x) = max(0, x) \tag{3.15}$$

The main issue of using ReLU in CNN is that it immediately set the negative values in the feature maps to zero which decreases the CNNs to train or learn from the input data properly.

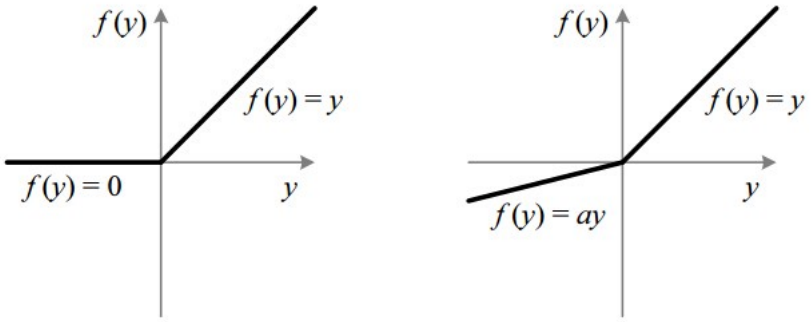


FIGURE 3.14: This image shows ReLU (left) activation vs Leaky ReLU (right), ReLU set all the negative values to zero, where Leaky ReLU allows negative values

The solution of this problem is to use the Leaky ReLU which does not set the negative values to zero straightway as shown in Fig. 3.14.

3.2.1.3 Pooling Layer

Pooling layer is used in CNN to down-sample input feature map to reduce the number of parameters and computation. The Pooling Layer operates independently on every depth slice of the input feature map and resizes it spatially, using the MAX operation in Max pooling layer and AVG in Average pooling layer. The most common Max pooling layer used in CNN is a filter of 2×2 with a stride of as illustrated in Fig 3.15. The down-sampling of the sample input DFU image with pooling layer is shown in Fig. 3.16.

3.2.1.4 Fully Connected Layers

Fully connected layers can only take single dimensional data, and each neuron connects to all values in the previous layer. Each connection has weight applied, with basic matrix operation with the addition of a bias value. These layers are commonly referred to as the fully connected layers. As the input data to CNN for our work is images, which consists of two or more dimensions based upon the channels available in the final layer, matrix reshaping needs to be done. As the matrix is reshaped, the network loses the spatial information of input data.

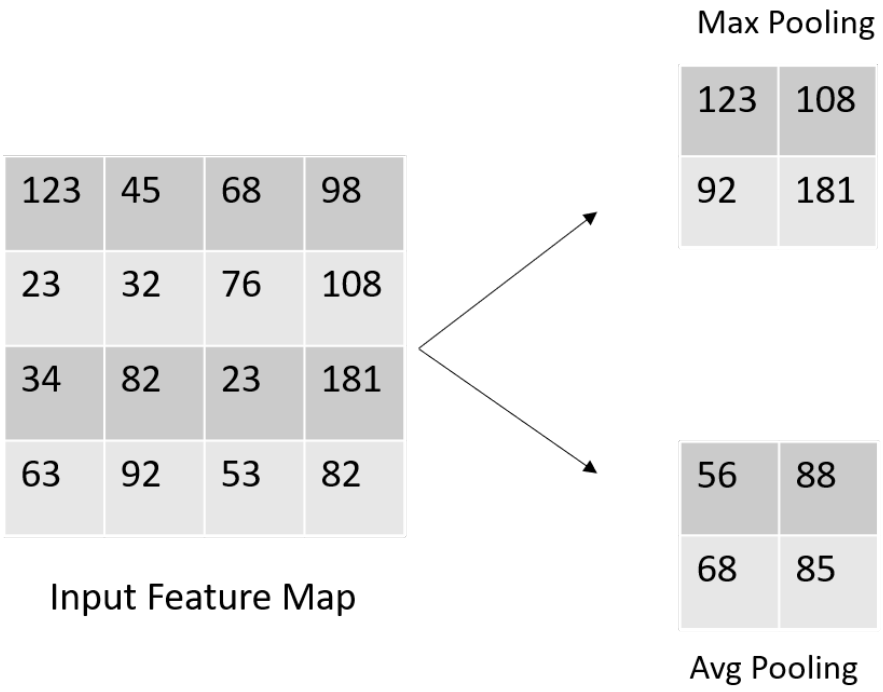


FIGURE 3.15: An example of a Max-pooling and Avg Pooling operation with filter size of 2×2 with a stride of 2 on input feature map.

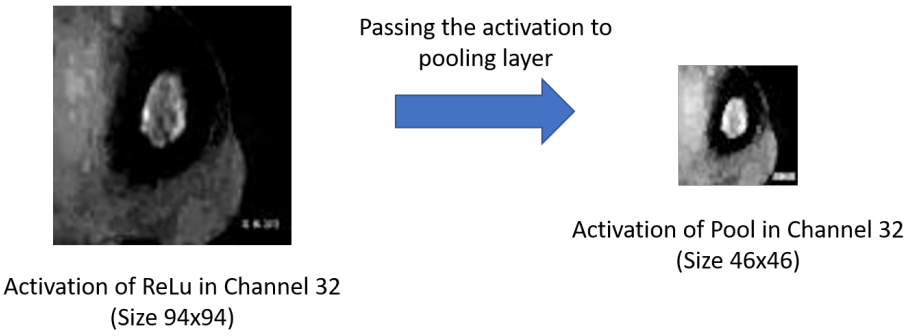


FIGURE 3.16: The example of activation of pooling layer in channel 32 of AlexNet on sample DFU image

3.2.1.5 Output

The output layer can be deemed as the last layer of the CNN. For classification, softmax is popularly used as an output layer in the CNN. The softmax output of class probabilities and is a measure of how close the parameters are with respect to the ground truth labels of the training and validation data. The softmax function (cross-entropy regime) is the final layer and is defined as

$$f_i(y) = \frac{e^{y_i}}{\sum_k e^{y_k}} \tag{3.16}$$

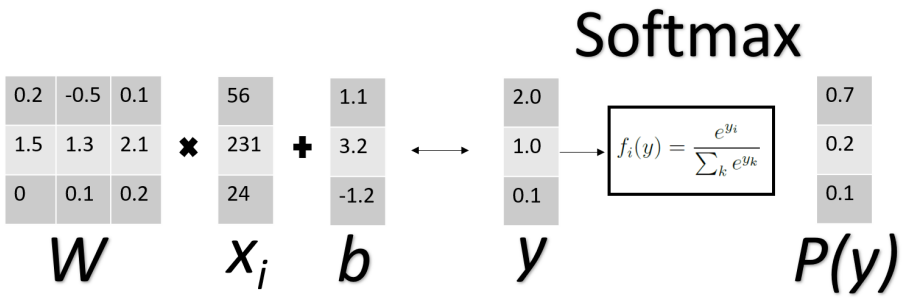


FIGURE 3.17: The example of converting the class scores by softmax function

where f_i is the i-th element of the vector of class scores f and y is a vector of arbitrary real-valued scores that are squashed to a vector of values between zero and one that sum to one.

Further explanation of how softmax is used to convert the class scores into probabilities is shown in Fig.

3.2.2 Loss Function

The loss is the error CNN makes during training while predicting the labels for the training and validation data. The loss is used in determining the effectiveness of CNN by evaluating how good (or bad) are the predicted probabilities. A good Loss function should return high values for bad predictions and low values for good predictions.

For binary classification (DFU skin and healthy skin), the commonly used loss function is binary cross entropy or log loss. The binary cross-entropy loss/ log loss is defined by

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \cdot \log\left(p\left(y_i\right)\right) + \left(1 - y_i\right) \cdot \log\left(1 - p\left(y_i\right)\right) \tag{3.17}$$

where y is the true label and p(y) is the predicted probability given by the CNN.

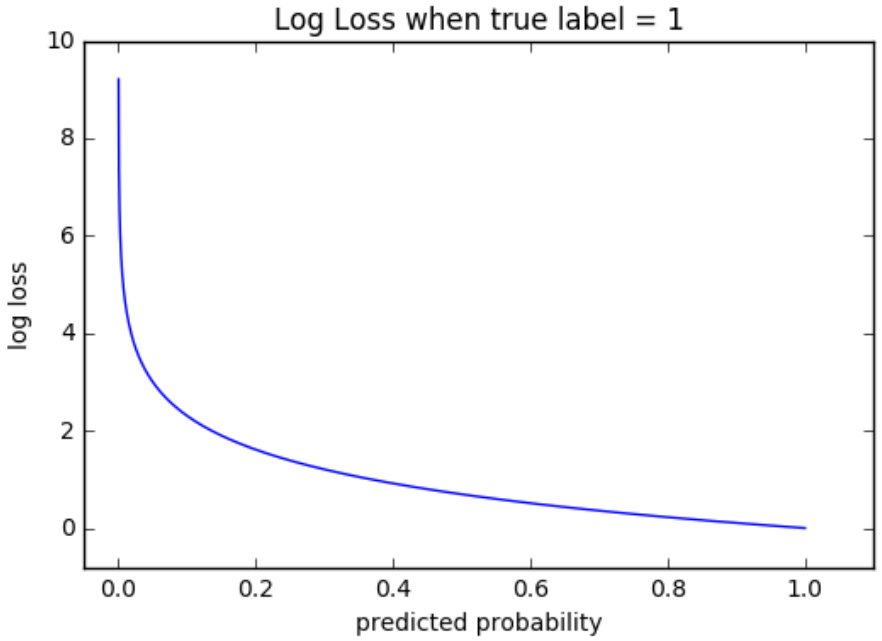


FIGURE 3.18: The example of log loss graph between the predicted probability and true label = 1)

As shown in Fig. 3.18, when the predicting probability is between 0 and 0.1, the log loss results in really high value, but as predicting probability approaches 1, the loss starts to decrease.

3.2.3 Optimisers

Optimisers are used to update the weights according to the loss in the training stage of the network; The purpose of the optimiser is to guide the weights associated with layers to minimize the loss while predicting the labels for the training and validation data.

A CNN works by combining different layers into one complete network. A standard CNN network takes training images as input. CNNs generally require a large amount of labelled training data to ensure high accuracy results. The network learns by processing the training data in the number of batches; larger batches allow the network to become more generalisable. Each batch of training images are processed by the network with the loss being calculated at each step, this is forward propagation. The next stage is backpropagation in which optimiser adjusts the weights of the neural network based upon the loss and learning rate value. Setting up a good learning rate is also a very important trick to train the

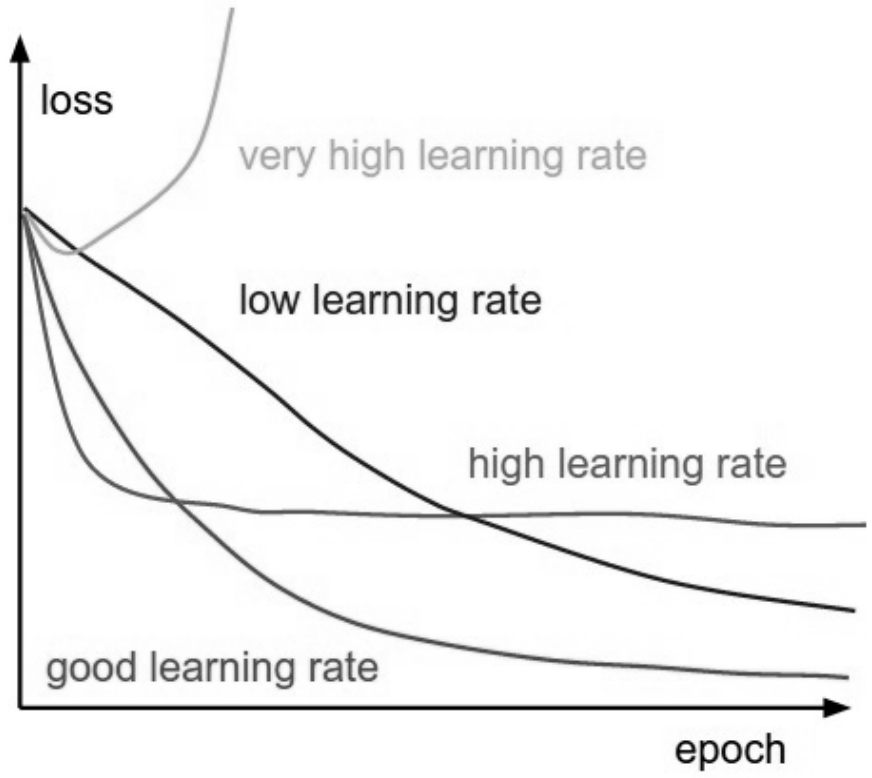


FIGURE 3.19: The good learning rate which is not high and really low trains CNN well

CNN effectively. Generally, small learning rate works very well for training the CNN rather than high learning rate as shown in Fig. 3.19.

The commonly used optimiser in CNN is Stochastic Gradient Descent (SGD) which is an extension of popular gradient descent. The gradient descent considers all the images in the training set to calculate the gradient in a single epoch. If the training set consists of thousands or millions of images, it would take a very long time to calculate the gradient descent for a single epoch. The SGD is an estimate of gradient descent of a small random sample from a training set.

Another optimizer method Adam [111] or Adaptive Moment Estimation performs very well in this field. Adam is an improved method that learns from previous methods, such as AdaDelta, by remembering previous gradients.

3.2.4 Cross-validation

In medical imaging, the cross-validation technique is popularly used to test the whole image dataset. Before we start to use any machine learning algorithm, the whole dataset is divided into the k-fold cross-validation data. For example, in 5-fold cross-validation, we would split the training data into 5 equal folds, use 3 and half of them for training, half for validation, and one for the testing set. We would then iterate over which fold is the testing fold, evaluate the performance, and finally average the performance across the different folds.

3.2.5 Batch Size, Epoch and step

A step indicates the processing of one batch of training images by CNN whereas an epoch indicates one iteration over all the images in the training set is processed. For example, if we have 2000 images in the training set, using a batch size of 100 means one epoch should contain 2000/100=20 steps.

3.2.6 Normalization

Before we input image data in CNN, the image data is normalized to reduce the computation and improving training time. In RGB images, the pixel values range from 0 to 255, these values are generally normalized by dividing 255. This results in pixel values range from 0 to 1. Another popular normalization technique is called zero-centering technique in which the mean of pixel values lies on the zero. It is done by subtracting pixel values with an overall mean value of pixels.

3.2.7 Transfer Learning

CNNs requires a considerable dataset to learn the features to get the positive results for the recognition of objects in images [100]. It is vital to use transfer learning from massive datasets in non-medical backgrounds such as ImageNet, Pascal-VOC, and MS-COCO dataset to converge the weights associated with each convolutional layers of network [25, 112, 113] for training the limited dataset. We utilized this transfer learning technique for training deep learning models for DFU segmentation and DFU localization. The main reason for using two-tier

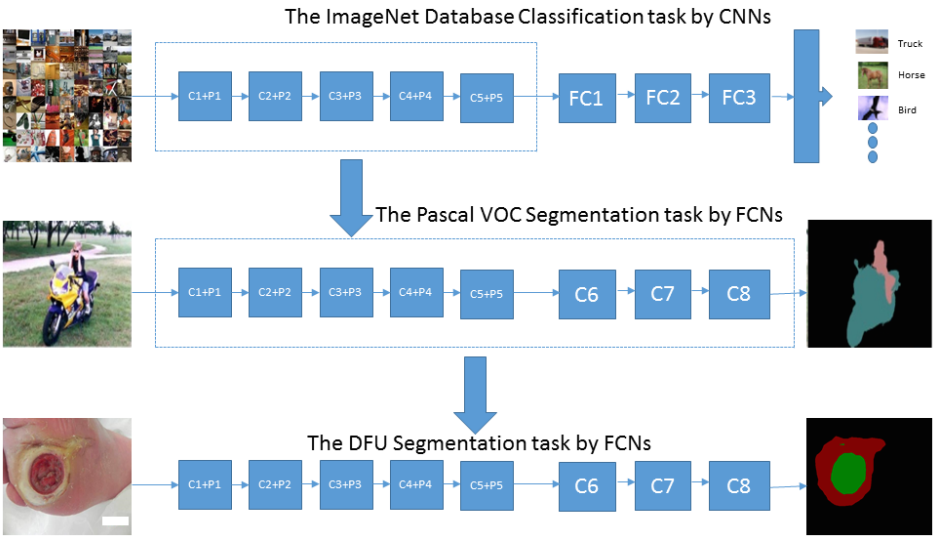


FIGURE 3.20: The two-tier transfer learning from big datasets to produce more effective segmentation

transfer learning in this work is because the medical imaging datasets are very limited. Hence, when CNNs are trained from scratch on these datasets, they do not produce effective results. There are two types of transfer learning i.e. partial transfer learning in which only the features from few convolutional layers are transferred and full transfer learning in which features are transferred from all the layers of previous pre-trained models. In both types of transfer learning, we can also freeze the features of layers. But, since we are transferring the features from non-medical datasets to medical datasets, we opted not to freeze any layers. We used both types of transfer learning known as two-tier transfer learning [25]. In the first tier, we used partial transfer learning by transferring the features only from the convolutional layers trained on most significant classification challenge dataset called ImageNet which consists of more than 1.5 million images with 1000 classes [3]. In the second tier, we used full transfer learning to transfer the features from a model trained on image segmentation dataset called Pascal VOC that consists of 2913 images of 21 classes for DFU segmentation task. For DFU localization task, we utilized the object localisation dataset called MS-COCO that consists of more than 80000 images with 90 classes [5]. The framework of two-tier transfer learning is demonstrated in Fig. 3.20. These two-tier transfer learning pre-trained models are used for training the deep learning models on DFU dataset results in better convergence of weights rather than initialising the weights randomly.

3.3 Summary

All of the theories and techniques regarding image processing, machine learning, CNN described in this Chapter 3 that provides the foundation on which the following contribution Chapters will be based. Chapter 4 describes the different DFU datasets and expert annotations used to perform different computer vision tasks for identification of DFU. Chapter 5 describes the classification models based on traditional machine learning and deep learning to classify normal skin and abnormal skin DFU of foot region in the dataset of 292 images. Chapter 6 introduces deep learning segmentation models to segment both DFU and surrounding skin in the foot images in the DFU dataset of 600. Chapter 7 outlines the robust methods for the localization of DFU in the extensive dataset of 1775 DFU images and transfer these models on the mobile devices for real-time detection of DFU. The final contribution in Chapter 8 outlines the machine learning methods to detect infection and ischemia in DFU with natural data-augmentation performed on 1459 foot images of DFU dataset.

Chapter 4

DFU Dataset and Performance Metrics

This Chapter focuses on DFU dataset with expert annotations is described in terms of classification, segmentation and localisation. Further section in this chapter is dedicated to the performance measures used in the different computer vision tasks for DFU recognition

4.1 DFU Dataset and Expert Labelling

To demonstrate the potential of this experiment, we utilized two different types of DFU dataset that are standardised dataset and non-standardised dataset for different medical imaging tasks. In the non-standardised dataset, we have collected 1500 patient's foot with DFU over the previous ten years at the Lancashire Teaching Hospitals, obtaining ethical approval from all relevant bodies and patient's written informed consent for the purpose of teaching and learning. These DFU images were captured with different cameras that are Nikon D3300, Kodak DX4530, and Nikon COOLPIX P100. In this dataset, the images are captured with inconsistent angles and orientation. Whereas in the standardised dataset, foot images are captured by IPad with FootSnap mobile application to show the robustness of algorithms over heterogeneous capture setup. It consists of the feet of 15 people with diabetes, aged between 43 and 74, and 15 non-diabetic control volunteers [64]. This dataset consists of 120 images that include both 105

(a) (b)

(c) (d)

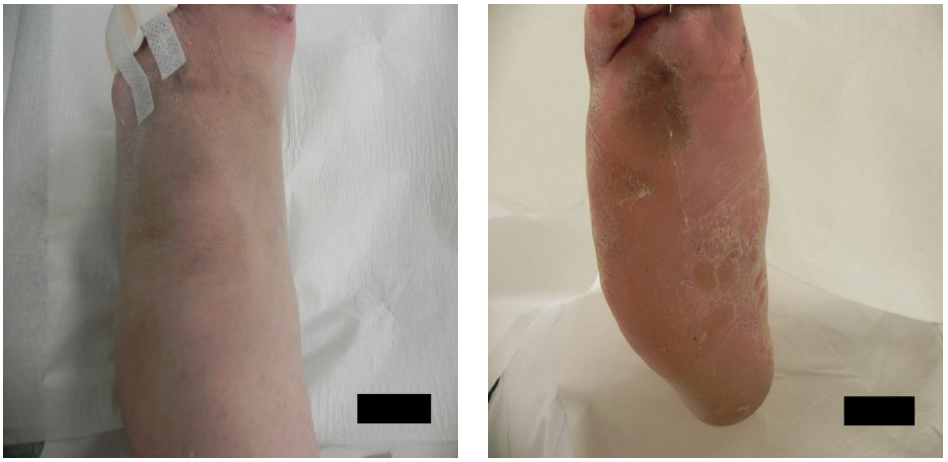
FIGURE 4.1: (a) and (b) are examples of non-standardised dataset (c) and (d) are examples of non-standardised dataset

healthy foot and 15 DFU foot images. Few examples of the non-standardised and standardised dataset are demonstarted in Fig. 4.1.

In this dataset, we excluded the cases such as out of focus, leg ulcer, no visible DFU from 1500 images as shown in Fig. 4.2.

There is no metadata regarding the patient's age, identity, sex, conditions of DFU such as infection, ischemia, depth, area, site included in this dataset. The main focus of this work is expert labelling of DFU according to the popular medical imaging tasks on this DFU dataset. The ground truth was produced by three healthcare professionals (two podiatrists and a consultant physician with specialization in the diabetic foot) specialized in diabetic wounds and ulcers. Where

Out of focus cases



No visible DFU

Leg Ulcer Cases

FIGURE 4.2: Types of images excluded for this experiment

there was disagreement, the final decision was made by the main podiatrist. We obtained the different number of expert labelling for DFU dataset for different computer vision tasks.

4.1.1 Expert Annotations in DFU Classification

In DFU classification, we utilized a subset of dataset consists of 292 images of patient's foot with DFU. We also utilized a subset of standardised DFU dataset to show the robustness of algorithms over heterogeneous capture setup and also to get more patches for normal class. It consists of 20 abnormal skin patches and 32 normal skin patches in this heterogeneous test case.

With the available annotator from Hewitt et al. [4], for each full image of a foot with ulcers, the medical experts delineated the ROI which is an important region around the DFU comprises of significant tissues of both normal and abnormal skin. The ground truth labels are delineated by medical professionals in the form of both normal and abnormal skin patches from the ROI region. In the collection of ground truth patches, the experts only collected both classes of patches from ROI region that helped with a more robust classification of the patches rather than involving the whole foot as a region. For each delineated abnormal region, the ground truth of the type of the abnormality was labelled and exported to a Extensible Markup Language (XML) file. For the annotation of 397-foot images with both ulcer and non-ulcer, there is a total of 292 ROI (Only for the foot images with ulcers). From these annotations, we produce a total of 1679 skin patches with 641 of normal and 1038 of abnormal class. Finally, we divided the dataset into a training set of 1423 patches, validation set of 84 patches and testing set of 172 patches. The annotator tool which can delineate the image into different types of patches is shown in Fig. 4.3.

4.1.2 Expert Annotations in DFU Segmentation

A subset of the images was used for this study, which includes 600 DFU images and 105 healthy foot images. The ground truth annotation of our dataset was performed by a podiatrist specialising in the diabetic foot and validated by a consultant specialising in diabetes. We created ground truth for each image with DFU by using Hewitt et al. [4] annotator. For each DFU image (as illustrated in Fig. 4.4), the expert delineated the region of interest (ROI) as the combination of ulcer and its surrounding skin. Then in each ROI, the two classes were labelled separately and exported to an XML file. These ground truths were further converted into the label image of single channel 8-bit paletted image (commonly

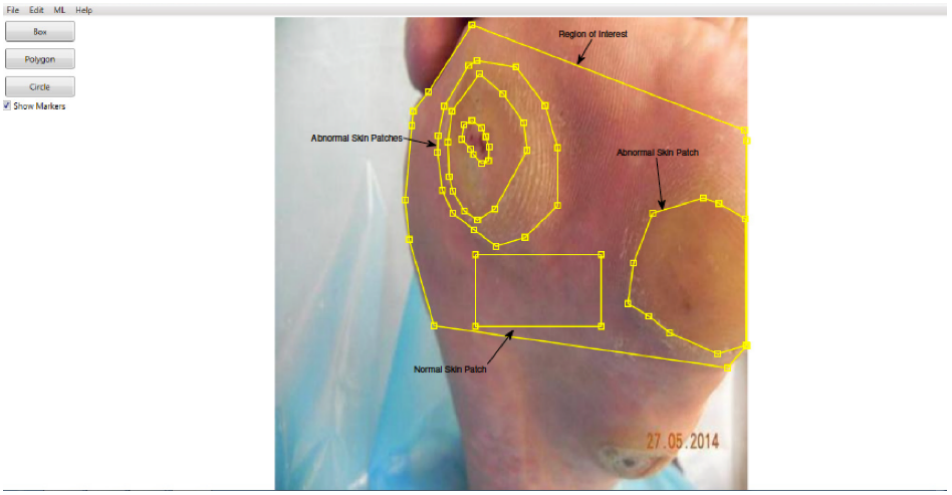


FIGURE 4.3: An example of delineating the different regions from the whole foot image to produce abnormal and normal skin patches with the help of annotator software [4].

FIGURE 4.4: An example of delineating the different regions of the pathology from the whole foot image and conversion to Pascal VOC format

known as Pascal VOC format for semantic segmentation) as shown in Fig. 4.4. In this format, index 0 maps to black pixels represent the background, index 1 (red) represents the surrounding skin and index 2 (green) as DFU. From 600 DFU images in our dataset, we produce 600 ROIs of ulcers and 600 ROIs for surrounding skin around the ulcers.

4.1.3 Expert Annotations in DFU Localisation

In the localisation experiment, we utilized the DFU dataset has a total of 1775 foot images with DFU. To test the specificity measure for the algorithms, we have included 105 healthy foot images in the DFU dataset from the FootSnap application [65].

In this dataset, the size of images varies between 1600×1200 and 3648×2736. We resized all the images to 640×640 to improve the performance and reduce

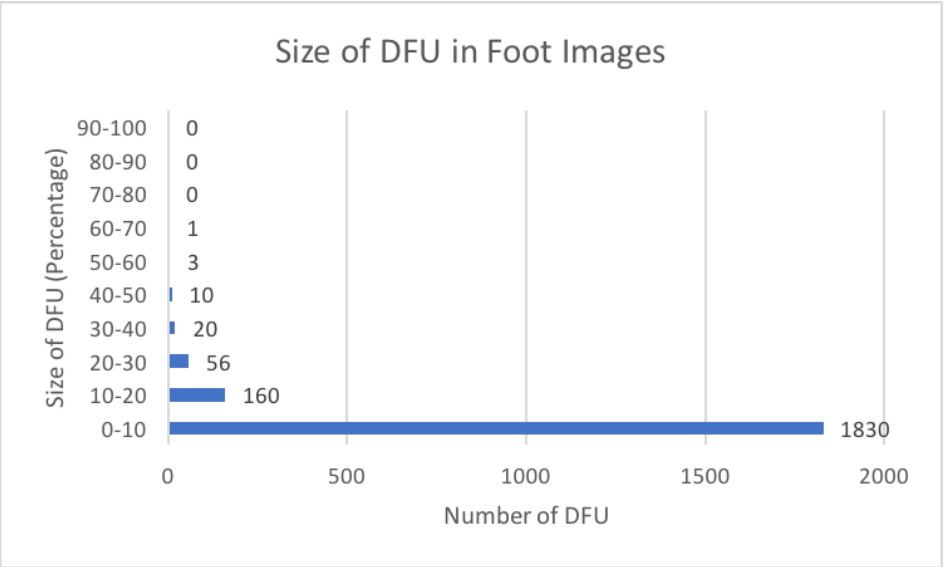


FIGURE 4.5: Comparison of Size of DFU against the size of image in the DFU dataset of 1775 images

the computational costs. We used Brett et al. [4] annotation tool for producing the ground truths in the form of a bounding box as shown in Fig. 4.6. In the DFU dataset, there is only one bounding box in approximately 90% images, two bounding boxes in 7% and finally, more than two bounding boxes in the remaining 3% images of the whole dataset. The medical experts delineated a total of 2080 DFUs (some images with more than one ulcer) using an annotator software. As shown in Fig. 4.5, approximately 88% DFU have the size of less than 10% of the actual size of an image. The size varied considerably across the DFUs in the dataset.

4.1.4 Expert Annotations for Recognition of Ischemia and Infection in DFU

For the recognition of ischemia and infection in DFU according to the Sinbad medical classification systems, we utilized dataset of 1459 images of patient's foot with DFU. Expert labelling of DFU to determine the binary classification of infection and ischemia on this DFU dataset is really important for this task. For this task, we performed the separate binary classification of ischemia and infection rather than performing combined infection and ischemia classification because of an unbalanced dataset as shown in Fig. 4.7 especially in the category where ischemia is present in DFU without infection (Only 24 cases).

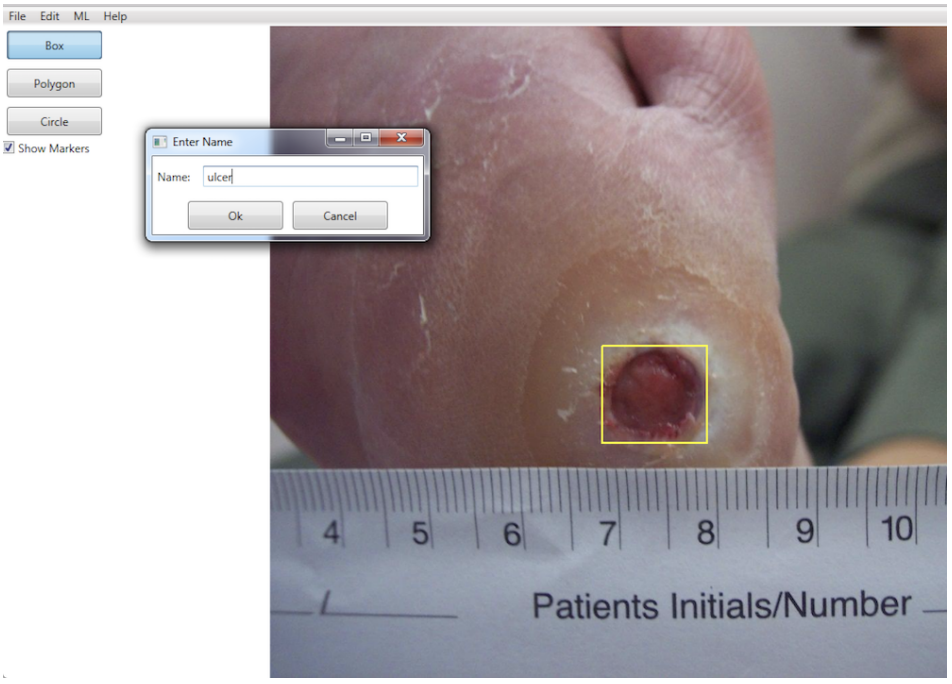


FIGURE 4.6: Annotation of ground truths on foot images for DFU localization

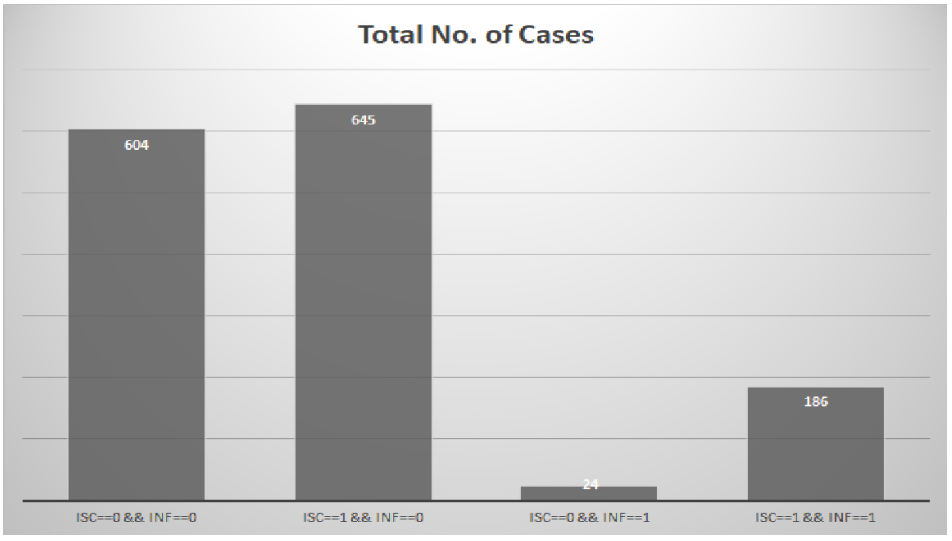


FIGURE 4.7: Comparison of combined Ischemia and Infection cases in the DFU dataset where ISC stands for ischemia and INF is infection

TABLE 4.1: The total number of cases of each condition of DFU

Category	Definition	No. of Cases	No. of DFU patches	No. of Augmented patches
Ischemia	Pedal blood flow intact	1249	1431	4935
	Clinical evidence of reduced pedal blood flow	235	705	4935
Total images		1459	1666	9870
Bacterial infection	None	628	684	2946
	Present	831	982	2946
Total images		1459	1666	5892

The complete number of cases of expert annotation of binary classification of ischemia and infection is detailed in Table 4.1. As shown in Table 4.1, the number of cases for ischemia and no ischemia in DFU are quite unbalanced whereas infection and no infection cases are fairly balanced. To balance the dataset, we used the natural data-augmentation technique [58].

4.2 Performance Measures

To measure the performance of the classification algorithm, quantification of results will be presented by various evaluation metrics such as *Accuracy*, *Sensistivity*, *Specificity*, *Recall*, *Precision*, *F-Measure* and *Matthews Correlation Coefficient* (MCC). These measurements are commonly used for binary classification purposes, and so is adequate for quantifying True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN) detections.

4.2.1 Accuracy, Precision, Sensitivity and Specificity

Accuracy is a measure of correctness of classifier. But, if data is unbalanced, then *Accuracy* is not reliable performance measures. By using the *Precision* measure of exactness, and determines a fraction of relevant responses from results. In medical imaging, both *Sensitivity* and *Specificity* are considered as very important metrics. *Sensitivity*, or *Recall* is a fraction of the results that are relevant to the experiment and that are successfully retrieved. *Specificity* determines the classifier's ability to identify negative results. In medical imaging, specificity of the test is the

proportion of patient's medical image that do not to have abnormality and will successfully test negative for it.

$$Accuracy = \frac{TP + FN}{TP + TN + FP + FN} \tag{4.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{4.2}$$

$$Sensitivity = \frac{TP}{TP + FN} \tag{4.3}$$

$$Specificity = \frac{FP}{FP + TN} \tag{4.4}$$

It is unlikely to use these measures on their own as both these measures are commonly used together to form an understanding of the relevance of the results returned from experimental classification.

4.2.2 F-Measure

The F-Measure is useful in determining the harmonic mean between the *Precision* and *Recall* and is used in place of accuracy as it provides a more detailed analysis of the data. The equation can be defined as

$$F\text{-}Measure = \frac{2TP}{2TP + FP + FN}. \tag{4.5}$$

A downside to this measure is that it does not take into account TNs, a value that is required to create Receiver Operating Characteristic (ROC) curves.

4.2.3 Matthews Correlation Coefficient

The Matthew's Correlation Coefficient (MCC) uses all detection types to output a value between -1, which indicates total disagreement and $+1$, which indicates total agreement. A value of 0 would be classed as a random prediction, and therefore both variables can be deemed independent. It can be provide a much more balanced evaluation of prediction than previous measurements, however it

is not always possible to obtain all four detection types (TP, FP, FN, TN). The coefficient can be calculated by

$$\text{MCC} = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4.6}$$

4.2.4 ROC Curve and AUC

An ROC curve determinnes the performance of a classifier at all classification thresholds which include two parameters True Positive Rate (TPR) and False Positive Rate (FPR). It plots TPR vs FPR at different classification thresholds. Area Under the ROC Curve (AUC) measures the entire 2D area underneath the entire ROC curve which provides an aggregate measure of performance across all possible classification thresholds.

$$TPR = \frac{TP}{TP + FN} \tag{4.7}$$

$$FPR = \frac{FP}{FP + TN} \tag{4.8}$$

4.2.5 Performance Measures in Segmentation

The performance measures for segmentation algorithms are almost similar to the classification measures. Additionally, Jaccard Similarity Index (JSI) and dice are popularly used by the researchers for the segmentation tasks. Dice is same as of F-Measure and is more forgiving than JSI in comparing the similarity of the shapes (Output and ground truth).

$$\text{JSI} = \frac{TP}{TP + FP + FN}. \tag{4.9}$$

$$Dice = \frac{2TP}{2TP + FP + FN}. \tag{4.10}$$

4.2.6 Performance Measures in Localization

All performance metrics are calculated with "overlap criterion" as an Intersection over Union (IoU) of the detected lesion and GT. A TP is when the IoU > 0.5. A FP is a detected ROI with IoU ≤ 0.5 and the duplicate bounding boxes. A FN is when there is no ROI detected by the algorithm.

We evaluate the performance of the proposed methods using three metrics, i.e. Precision, Recall and Mean IoU. The Precision is calculated by the number of TP divided by the sum of the number of TP and FP. The Recall is the number of TP divided by the sum of a number of TP and FP. We also report the Mean IoU, which is the average of the overlap percentage of the TP cases (detected DFU).

4.3 Summary

In this chapter, we discussed the different DFU datasets and expert annotations used to perform different computer vision tasks for recognition of DFU. One of the major focus of this thesis to prepare the DFU datasets and expert annotations according to the popular machine learning and deep learning libraries such as Caffe, Tensor-flow, PyTorch. We received the expert annotations from the experienced podiatrists for classification, segmentation and localization in the XML format using in-house annotator. We cleaned the dataset and converted the format of these expert annotation according to the input format of ground truths required by the deep learning methods such as Pascal VOC format for segmentation. In the later section, the brief discussion of popular performance metrics used in our experiments is provided.

Chapter 5

DFU Classification

This Chapter presents preliminary investigations of DFU classification problem. We assessed the two classes as normal skin (healthy skin) and abnormal skin (DFU). This work investigated the use of machine learning algorithms to extract the features for DFU and healthy skin patches to understand the differences in the computer vision perspective

5.1 Introduction

In this work, we proposed computer vision algorithms to differentiate DFU from healthy skin with traditional machine learning and deep learning approaches. The main motive of performing this 2-class classification was to find the misclassified cases of both DFU skin and healthy skin patches by the classifier. Misclassified cases, especially in DFU skin patches, could help us understand which DFU of particular stages/grades are not well recognised by the computer vision techniques. The key contributions of this work include:

1. This work presented the related computerised telemedicine systems designed for DFU. We also presented the DFU dataset of 397 foot images, which consists of 292 images with DFU and 105 healthy foot images. The podiatrists delineated the total of 1679 skin patches with 641 of healthy skin and 1038 of DFU.

2. To the best of our knowledge, this is the first time, machine learning algorithms are used to understand and extract the computer vision features from DFU and healthy skin patches. We used CNNs to develop a fully automatic method to classify the DFU skin against the normal skin.

3. Development of a novel CNN architecture called DFUNet, which was fine-tuned to process the input data more effectively and efficiently than other comparative state-of-the-art CNNs architecture. CNNs require substantial data to produce very accurate results, but with the help of larger filter sizes in the blocks of convolutional layers in parallel, it can produce good results on small dataset such as DFU and facial skin dataset.

5.2 Methodology

This section describes the feature descriptors used in experiments, including for CML, the CNNs architecture of LeNet, AlexNet, and GoogLeNet. Finally, we proposed our own CNN architecture, DFUNet, to improve the way DFU are classified.

5.2.1 Data Augmentation of Training Patches

Deep networks require a lot of training image data because of the enormous number of parameters, especially weights associated with convolutional layers needed to be tuned by learning algorithms. Hence, we used data augmentation to improve the performance of deep learning methods. We used the combination of various image processing techniques like rotation, flipping, contrast enhancement, using different color space, and random scaling to perform data augmentation. The rotation was performed by rotating the image by the angle of 90°, 180°, 270°. Then, three types of flipping (horizontal flip, vertical flip and horizontal+vertical flip) performed on the original patches. The four color space that are used for data augmentation are *Ycbcr*, *NTSC*, *HSV* and *L*a*b*. In the contrast enhancement, we used the three functions called adjust image intensity value, enhanced contrast using histogram equalization, and contrast-limited adaptive histogram equalization. We produced the two times cropped patches with the help of random offset

and random orientation from the original dataset of skin patches. With these techniques, we increased the number of training and validation patches by 15 times i.e. 21,345 patches for training and 1260 patches for validation.

5.2.2 Pre-processing of Training Patches

Since we obtained a large number of training data with the help of data augmentation, it was essential to perform pre-processing on these patches. We used the zero-centring technique for pre-processing of these obtained patches, by subtracting pixel values with an overall mean value of pixels. So, that mean of pixel values lies on the zero.

5.2.3 Conventional Machine Learning

We investigated the use of human design features with CML on DFU and healthy skin classification. From our observation on the differences between DFU and healthy skin, the color and texture feature descriptors were the visual cues for classification. For this 2-class classification problem, the Sequential Minimal Optimization (SMO) [114] was selected as SVM based machine learning classifier.

5.2.4 Convolutional Neural Networks

For comparison with the traditional features, deep learning, specifically convolutional neural networks, were used to classify between healthy foot skin and skin with diabetic ulcerations. The first architecture, we used was LeNet [115] running for 60 epochs, a learning rate of 0.01 with a step-down policy and step size of 33%, and gamma is set to 0.1. This network was originally used for recognizing digits and zip codes. These simple structures are easily recognized, even in hand-written datasets such as MNIST [2].

Using LeNet represented these structures much better than traditional features, even on a relatively small training set of 1423 patches and validation of 84 patches.

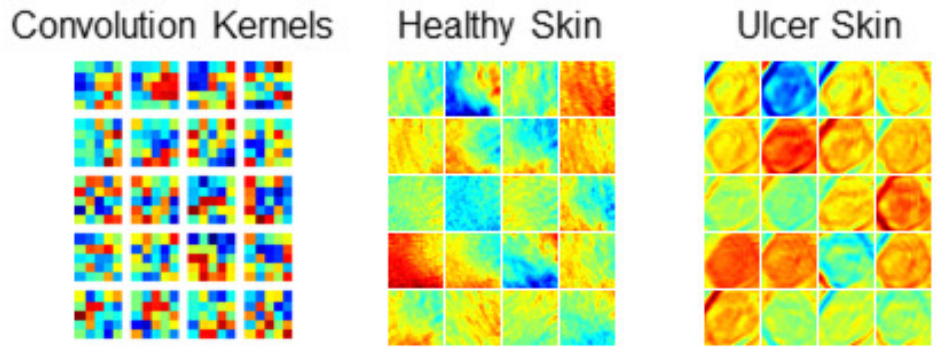


FIGURE 5.1: The output of healthy and diabetic ulcer skin from the first convolution layer of LeNet highlight discriminative features.

The input was 28×28 patches of skin in grayscale split into abnormal and normal skin samples. At the first convolution layer shown in Fig. 5.1, the kernels and activations already showed the effectiveness of CNNs when highlighting important features.

We used the Caffe [116] framework to implement LeNet [115], and used the Adaptive Moment Estimation (Adam) [117] method for stochastic optimisation. This solver combines the advantages found in AdaGrad [118], which works well with sparse gradients, and RMSProp [119], which works well in an online setting. Adam is intended for large datasets and variability in parameters. However, the results in Table 5.4 show that smaller datasets work as effectively.

We also used popular CNN model AlexNet for classification of abnormal (DFU) and normal (healthy skin) classes. This network was originally used for classification of 1000 different objects of classes on ImageNet dataset. It emerged as the winner of ImageNet ILSVRC-2012 competition in classification category by achieving 99% confidence. There are few adjustments made in the original network to work well for our 2-class classification problem. Also, a pre-trained model was used for better convergence of weights to achieve better results [3]. To train the model on Caffe framework, we used the same parameters as in LeNet i.e. 60 epochs, a learning rate of 0.01, and gamma of 0.1.

Another state-of-the-art CNN architecture that we used is GoogLeNet [17], a 22 layers deep network, with a similar experimental setting as of LeNet and AlexNet. Szegedy et al. [17] introduced a new module called inception to GoogLenet. This acts as a multiple convolution filter inputs, which are processed on the same input and also does pooling at the same time. All the outcomes are then merged

into a single feature layer. This layer allows the model to take advantage of multi-level feature extraction from each input. Again, a transfer learning approach using pre-trained models to improve performance.

5.2.5 Proposed Method - Diabetic Foot Ulcer Network

Since this experiment focused on the types of skin lesion which are at high risk of being misclassified by computer vision algorithms. ResNet [120], DenseNet [121], Inception [17] frameworks are very deep and computational intensive networks to work on this basic binary classification problem. The traditional CNNs such as AlexNet [3, 100] use only single type of convolutional filters popularly ranging from 1×1 to 7×7 on the input data. To improve the extraction of important features for DFU classification, we proposed a new Diabetic Foot Ulcer Network (DFUNet) architecture which is inspired from GoogleNet with two major interventions that are the depth of the network and filter sizes in the block of convolution layers in parallel. For DFUNet, we significantly decreased the depth of the network from 22 layers to 14 layers, but, the number of filters in the block of convolutional layers in parallel is increased significantly to learn more features maps. Since deeper CNNs convolve more input data, it also leads to a large number of parameters and computation time. Since there are discriminative feature difference between healthy and DFU skin, we used less number of layers and increased the filter sizes in DFUNet to reduce the computation. DFUNet combines two types of convolutional layers i.e. traditional convolution layers at the starting of the network which use blocks of single convolutional layer followed by blocks of convolutional layers in parallel, which use multiple convolutional layers in parallel for extraction of concatenated features from the same input. We tested different variants of DFUNet by using the different number of filter sizes in the block of convolutional layers in parallel. Detecting changes in healthy skin is a clear computer vision problem similar to malignant skin lesions, so the DFUNet is designed around convolutions to finding discriminative features for learning.

Healthy skin tends to exhibit smooth textures and DFU have many distinct features including long edges, sharp changes in intensity or color and quick changes between surrounding healthy skin and the DFU itself. DFUNet, summarised in Fig. 5.2, is split into three main sections: the initialisation layers inspired by GoogLeNet, blocks of convolution layers in parallel to discriminate the DFU more

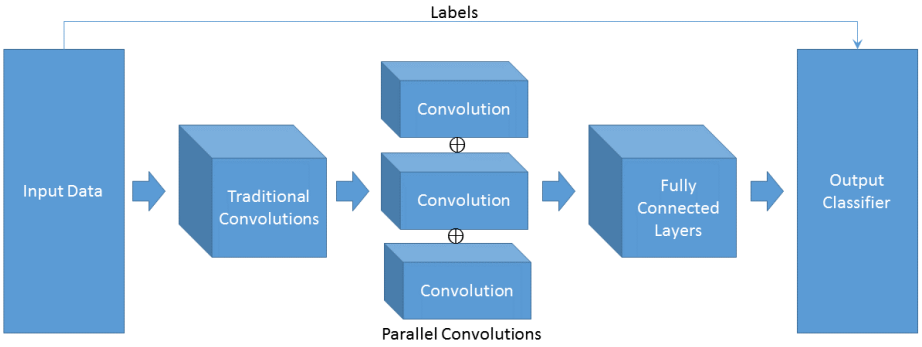


FIGURE 5.2: An overview of the proposed DFUNet architecture. The proposed DFU architectures consists of Input Data block which consists of training and validation data, Traditional Convolution block consist of single convolutional layers, block of convolutional layers in parallel to extract concatenated features with the help of different convolutions, Fully Connected layers which act as neural network and finally, Output Classifier to produce the prediction of class label

TABLE 5.1: Complete description of Network Architecture of DFUNet. Conv. refers to convolutional layer, Max-pool. refers to Max-Pooling layers. There are variations in filter size of blocks of convolutional layers in parallel of different variant of DFUNet.

Layer no.	Layer type	Filter size	Stride	No. of filters	FC units	Input	Output
Layer 1	Conv.	7×7	2×2	64	-	3×224×224	64×112×112
Layer 2	Max-pool.	3×3	2×2	-	-	64×112×112	64×56×56
Layer 3	Conv.	1×1	1×1	64	-	64×56×56	64×56×56
Layer 4	Conv.	3×3	1×1	192	-	64×56×56	192×56×56
Layer 5	Max-pool.	3×3	2×2	-	-	192×56×56	192×28×28
Layer 6	Conv. in parallel	1×1,3×3,5×5	1×1	32⊕64⊕128	-	192×28×28	224×28×28
Layer 7	Max-pool.	3×3	2×2	-	-	224×28×28	224×14×14
Layer 8	Conv. in parallel	1×1,3×3,5×5	1×1	32⊕64⊕128	-	224×14×14	224×14×14
Layer 9	Conv. in parallel	1×1,3×3,5×5	1×1	32⊕64⊕128	-	224×14×14	224×14×14
Layer 10	Max-pool.	3×3	2×2	-	-	224×14×14	224×7×7
Layer 11	Conv. in parallel	1×1,3×3,5×5	1×1	32⊕64⊕128	-	224×7×7	224×7×7
Layer 12	Max-pool.	7×7	1×1	-	-	224×7×7	224×1×1
Layer 13	Fully conn.	-	-	-	224		
Layer 14	Fully conn.	-	-	-	No. of Classes		

efficiently than previous network layers and lastly, both fully-connected layers and a softmax-based output classifier. The detailed layers of the general DFUNet architecture are provided in Table 5.1.

The parameters used for training with DFUNet are 60 epochs, a batch size of 8, the Adam solver with a learning rate of 0.001. A step-down policy was used where the learning rate reduced with a step of 33% and gamma was set to 0.1.

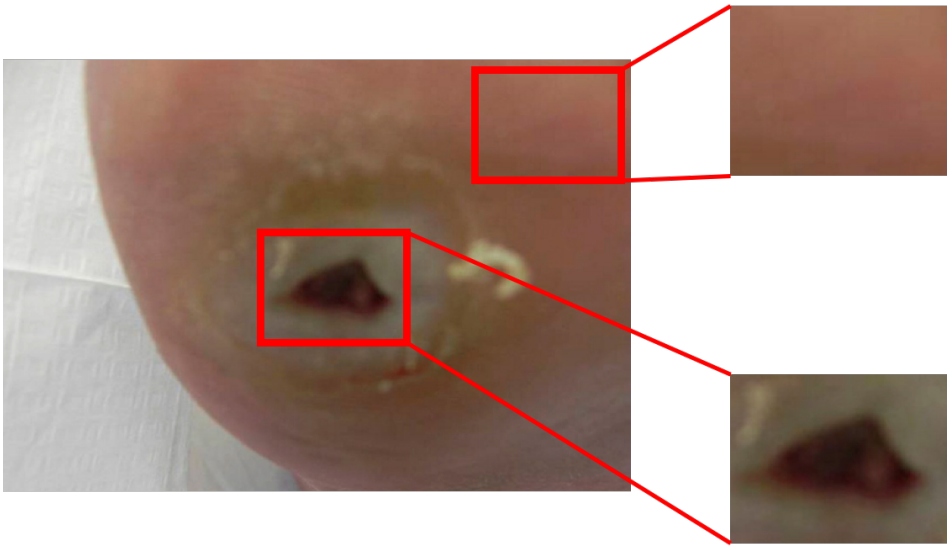


FIGURE 5.3: Healthy and ulcer patches taken from feet for training in the CNN.

The Configuration of GPU Machine for Experiments was: (1) Hardware: CPU - Intel i7-6700 @ 4.00Ghz, GPU - NVIDIA TITAN X 12GB, RAM - 32GB DDR4 (2) Software: Caffe .

5.2.5.1 Input Data

The DFU training and validation images were resized as 256×256 patches from areas of the feet containing DFU and healthy skin. An example of the regions of a foot cropped is shown in Fig. 5.3. We used the centre crop of size 224×224 and mirror as data parameters. Initial traditional convolutional block consists of single convolution filters at each step to reduce the computational cost on feature maps. Inspired by the GoogLeNet [17] input stem, the input to DFUNet, begins by initial convolutions, pooling and normalisation layers in a traditional CNNs structure from layer 1 to layer 5 in Table 5.1. Doing this step also ensures that the larger raw input image dimensionality is reduced before moving on to subsequent layers.

5.2.5.2 Block of Convolution Layers in Parallel

The idea behind using the block of convolutional layers in parallel is basically concatenation of multiple convolution filter inputs to allow the multiple-level feature extraction and cover more spread out clusters from the same input. The design of

TABLE 5.2: The descriptions of filter size in the block of convolutional layers in parallel of different variants of DFUNet. Conv. refers to convolutional layer and var. refers to variant.

Layers No.	DFUNet Var. 1	DFUNet Var. 2	DFUNet Var. 3	DFUNet Var. 4	DFUNet Var. 5
1st block of Conv. in parallel	128⊕256⊕512	192⊕256⊕512	128⊕128⊕128	192⊕192⊕192	256⊕256⊕256
2nd block of Conv. in parallel	128⊕256⊕512	192⊕256⊕512	128⊕128⊕128	256⊕256⊕256	256⊕256⊕256
3rd block of Conv. in parallel	128⊕256⊕512	192⊕256⊕512	256⊕256⊕256	256⊕256⊕256	512⊕512⊕512
4th block of Conv. in parallel	128⊕256⊕512	192⊕256⊕512	256⊕256⊕256	512⊕512⊕512	512⊕512⊕512

the convolutions is weighted towards creating as discriminative features as possible to highlight any DFUs in an image. Three sizes of convolution kernels are used in the block of convolutional layers in parallel of DFUNet throughout: 5×5, 3×3 and 1×1. 1×1 convolution layer is used in the block of convolutional layers in parallel to reduce the dimensionality of your input to large convolutions such as 3×3 and 5×5, thus keeping computations reasonable. These are processed in parallel to each other and finally concatenated. The core of DFUNet is the three blocks of convolutional layers in parallel and is shown in Fig. 5.4. Increasing filter sizes in the block of convolutional layers in parallel is a key innovation in methods appears to be in the architecture of the DFUNet. As this is the one of the most significant innovation, the DFUNet is experimented with different variants of these blocks of convolutional layers in parallel to get the optimal architecture. We investigated the different sizes of filters in the block of convolutional layers in parallel by making 5 variants to get the best variant based on the performance metrics in Table 5.2. We created these five variants to test the hypothesis whether increasing the size of filters improves the performance of DFUNet or not. These variants were tested on the DFU dataset and the results are provided below in Table 5.3.

Each convolution provides additional discriminative power. Lower activations are present in healthy skin samples shown in Fig. 5.5 due to the absence of skin abnormalities. Higher activations are present in skin with an ulcer as shown in Fig. 5.6 due to skin abnormality.

Each convolution layer uses a ReLU which is defined as

$$f(x) = \max(0, x) \tag{5.1}$$

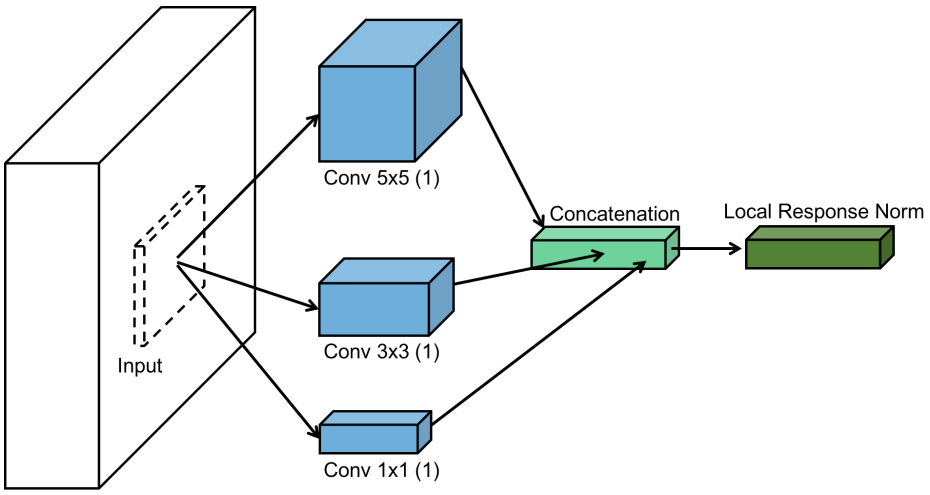


FIGURE 5.4: The structure of block of Conv. in parallel in which three types of convolutional filters are used, concatenation layers to concatenate the features of each convolutional filters, and finally pass it local response norm layer.

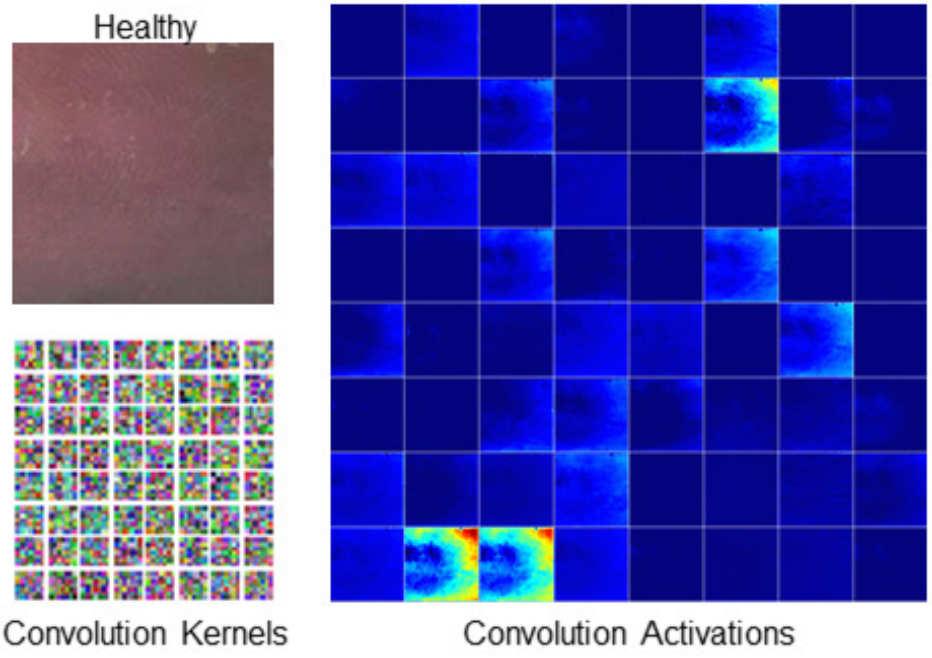


FIGURE 5.5: The convolution activation produced by the kernels of first convolutional layer on healthy skin raw input, to highlight the features learned by convolutional layer.

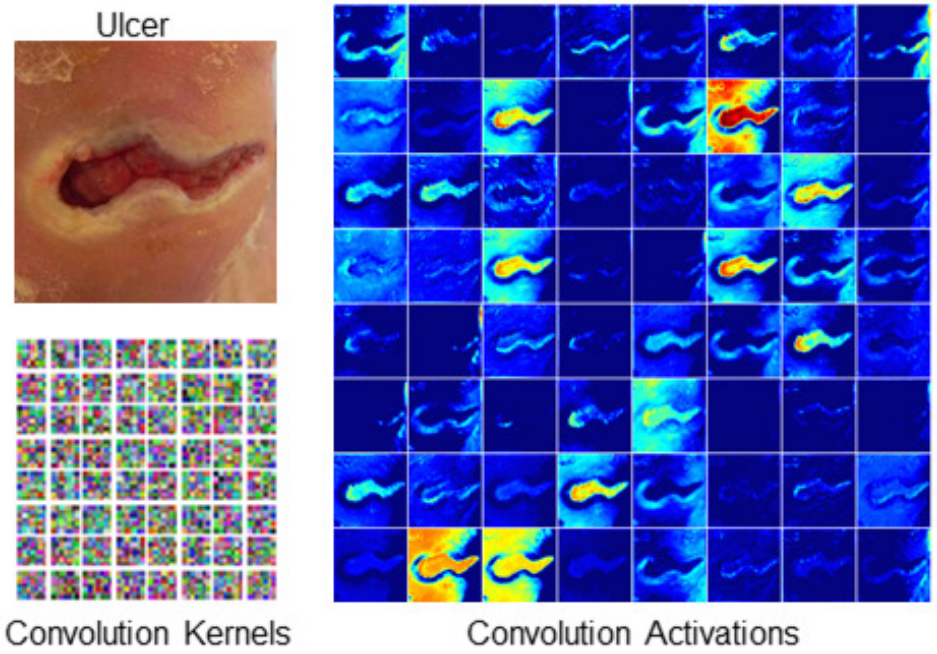


FIGURE 5.6: The convolution activation produced by the kernels of first convolutional layer on DFU skin patch, to highlight the discriminative features learned by convolutional layer.

where the function thresholds the activations at zero. As we used a ReLU for each convolution, they include unbounded activations, so we used Local Response Normalisation (LRN) to normalise these activations after each concatenation of convolutional layers. It is also proven helpful in avoiding the over-fitting problem faced by CNNs methods. Let, $a\frac{i}{x,y}$ be the source output of kernel i applied at position *(x,y)*. Then, regularized output $b\frac{i}{x,y}$ of kernel i applied at position *(x,y)* is computed by

$$b\frac{i}{x,y} = a\frac{i}{x,y}(k + \alpha \sum_{max(0,i-\frac{n}{2})}^{min(N-1,i+\frac{n}{2})} (a\frac{j}{x,y})^2)^\beta \tag{5.2}$$

where N is total number of kernels, n is the size of the normalization neighbourhood and α,β,k,(n) are the hyper-parameters.

Further, to reduce dimensionality, a max pooling layer is included after the first and the third block of convolutional layers in parallel.

5.2.5.3 Fully Connected Layers and Output Classifier

The final section is the softmax output of class probabilities and is a measure of how close the parameters are with respect to the ground truth labels of the training

and validation data. The 2-class outputs of the DFU are healthy skin and DFU. It is formed from an average pooling layer followed by two Fully Connected (FC) layers with outputs of 100 for the first and 2 for the second. It is worth mentioning, the computation complexity of DFUNet is further reduced for the 2-class problem by using only 100 neurons rather than 1000 in first FC layer and last FC layer is adjusted as 2. This modification in FC layers helps in faster processing time in both training and testing phase of the DFUNet. The softmax function (cross-entropy regime) is the final layer and is defined as

$$f_j(z) = \frac{e^{z_j}}{\sum_k e^{z_k}} \tag{5.3}$$

where f_j is the j-th element of the vector of class scores f and z is a vector of arbitrary real-valued scores that are squashed to a vector of values between zero and one that sum to one. The loss function is defined so that having good predictions during training is equivalent to having a small loss. The final layers including fully connected layers which work as a regular neural network which have connections to all activations in the previous layer, and softmax classifier, to predict the class label as either normal skin or DFU.

5.3 Results and Discussion

The DFU dataset was split into the 85% training, 5% validation and 10% testing sets and we adopted the 10-fold cross-validation technique. Hence, for training, validation, and testing set using the proposed DFUNet architecture, we used approximately 1423 patches (including 882 abnormal cases), 84 patches (including 52 abnormal cases), and 172 patches (104 abnormal cases) respectively from the 397 original foot images. As mentioned previously, we used both CML models and CNNs models to do the classification task. LeNet was the only architecture that worked on 28×28 grayscale patches rather than 256×256 RGB images as the input used by GoogLeNet, AlexNet, DFUNet and CML. It was included to show how the basic deep learning works on this new classification problem.

In Table 5.4, we report *Sensitivity*, *Specificity*, *Precision*, *Accuracy*, *F-Measure* and *Area under curve of ROC (AUC)* as our evaluation metrics. In medical imaging, *Sensitivity* and *Specificity* are considered reliable evaluation metrics for classifier completeness.

TABLE 5.3: The performance measures of various variants of the DFUNet on DFU dataset. where S.E. is standard error of AUC and C.I. is confidence interval of AUC curve

	Sensitivity	*Specificity*	*Precision*	*Accuracy*	*F-Measure*	*AUC Score*	*S.E.*	*95% C.I.*
DFUNet Var. 1	0.923±0.029	0.910±0.037	**0.946±0.021**	0.918±0.017	0.934±0.017	0.957	0.0049	0.9481 - 0.9673
DFUNet Var. 2	0.928±0.034	0.905±0.036	0.942±0.028	0.919±0.024	0.935±0.020	0.959	0.0046	0.9499 - 0.9678
DFUNet Var. 3	0.928±0.032	0.906±0.036	0.942±0.028	0.921±0.027	0.935±0.019	0.960	0.0045	0.9518 - 0.9694
DFUNet Var. 4	0.927±0.023	0.900±0.038	0.938±0.030	0.917±0.019	0.933 ± 0.017	0.958	0.0046	0.9496 - 0.9675
DFUNet Var. 5	**0.934±0.033**	**0.911±0.044**	0.945±0.032	**0.925±0.029**	**0.939±0.024**	**0.961**	**0.0044**	**0.9520 - 0.9695**

In Table 5.3, we report the performance measures of various DFUNet variants with different parameters as explained in the architecture of DFUNet in the previous section. There was not much gap in performances between all the models. But, overall, the DFUNet variant 5 performed best in every evaluation metrics except *Precision* in which DFUNet variant 1 performed the best. It also proved the earlier hypothesis correct as increasing the size of filters in the block of convolutional layers in parallel improved the performance of DFUNet. Hence, DFUNet variant 5 which uses the much larger filter sizes than other variants in the last two blocks of convolutional layers in parallel produced better results. Hence, with best results achieved by DFUNet variant, we used it as a proposed DFUNet to compare the performance with other traditional machine learning and deep learning models. ROC curve for all the variants is illustrated by Fig. 5.7.

There are three CML models and three CNNs models used for classification. In CML, we used the combination of LBP, HOG and Colour descriptors (*RGB*, *HSV* and *L*u*v*) as feature vectors and then, we trained an SMO for our classification problem. For each CNN, LeNet, AlexNet, GoogLeNet and our proposed DFUNet are the chosen architectures used for classification. Each classifier performed well for *Sensitivity* with less than 1.4% margin between the highest result (DFUNet) and the lowest result (LBP + HOG). There is a more significant gap of 7.7% in *Specificity* for the CML models performance measure, with results ranging from 0.835 to 0.845.

For the CNNs approaches, LeNet achieved the lowest score of 0.81 for *Specificity*, whereas the AlexNet, GoogLeNet and DFUNet performed best in this category, with 0.892, 0.912, and 0.908 respectively. AUC is considered to be a viable performance measure for the different machine learning approaches for classification, with DFUNet and GoogLeNet achieving 0.961 and 0.960 respectively.

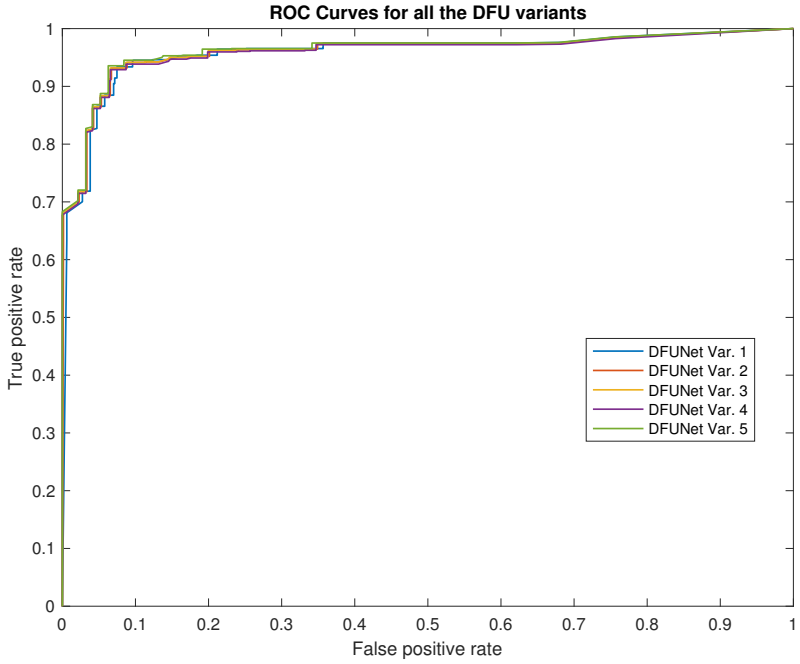


FIGURE 5.7: The ROC curve for all DFUNet models as mentioned in Table 5.3, DFUNet var. 5 performed best with an AUC score of 0.961. Var. refers to variant.

TABLE 5.4: The performance measures of binary classification task by both traditional machine learning and CNNs including our proposed method DFU-Net. Overall, our proposed DFUNet achieved the best results. where S.E. is standard error of AUC and C.I. is confidence interval of AUC curve

	Sensitivity	*Specificity*	*Precision*	*Accuracy*	*F-Measure*	*AUC Score*	*S.E.*	*95% C.I.*
LBP	0.919±0.029	0.764±0.052	0.878±0.038	0.865±0.038	0.898±0.033	0.932	0.0061	0.9202 - 0.9443
LBP + HOG	0.881±0.022	0.841±0.032	0.906±0.027	0.866±0.042	0.893±0.022	0.931	0.0060	0.9190 - 0.9427
LBP + HOG + Colour	0.902±0.027	0.845±0.027	0.904±0.025	0.880±0.034	0.904±0.024	0.943	0.0054	0.9324 - 0.9537
LeNet (CNN)[115]	0.912±0.026	0.810±0.063	0.871±0.038	0.872±0.041	0.893±0.019	0.929	0.0050	0.9405 - 0.9603
Alexnet (CNN)[3]	0.895±0.024	0.886±0.029	0.933±0.032	0.893±0.021	0.914±0.022	0.950	0.0050	0.9405 - 0.9603
GoogLeNet (CNN)[17]	0.905±0.027	**0.912±0.052**	**0.949±0.038**	0.907±0.022	0.927±0.019	0.960	0.0045	0.9514 - 0.9690
Proposed DFUNet	**0.934±0.033**	0.911±0.044	0.945±0.032	**0.925±0.029**	**0.939±0.024**	**0.961**	**0.0044**	**0.9520 - 0.9695**

Overall, we showed that using CNNs can outperform the more traditional CML features by a large margin. All CNN architectures achieved higher results than any of the CML results in most cases. GoogLeNet and DFUNet were the best performers for various evaluation metrics among all the classifiers. The ROC curve for all the models is demonstrated by the Fig. 5.8. The details of AUC performance for each method is described in Table 5.4.

We received better results than GoogLeNet on various evaluation metrics.

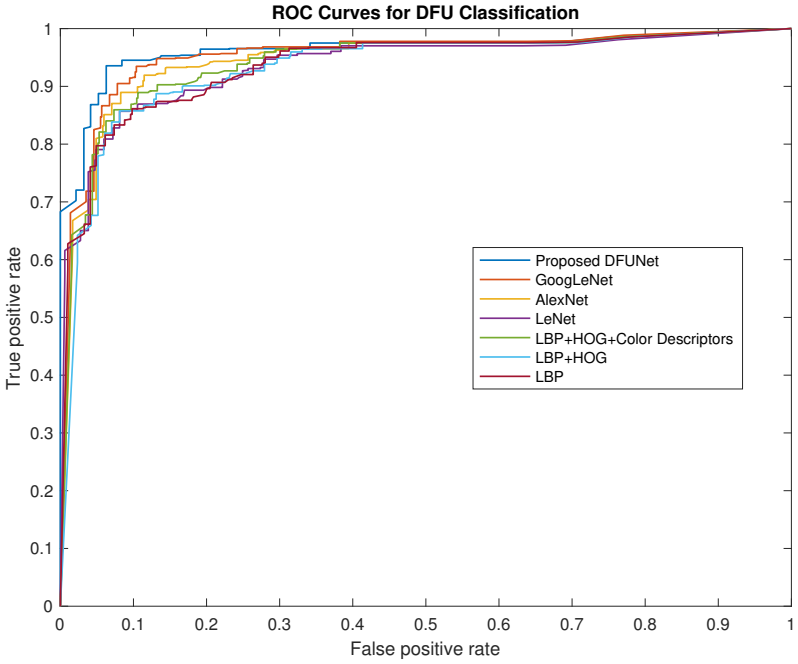


FIGURE 5.8: ROC curve for all the models including CML and CNNs mentioned in Table 5.4 in which our proposed DFUNet method achieved the best AUC score.

The reason behind using the DFUNet rather than conventional CNNs architecture, in particular, GoogLeNet is to speed up the best results with the help of lesser layers i.e. 14 layers architecture compared to the 22 layers architecture of GoogLeNet. We also reduced the number of neurons in the FC layers to improve the processing time of DFUNet according to the 2-class problem. With the 10-fold cross-validation, on the same machine configuration and input batch size on Caffe framework, DFUNet took an average of 3 minutes 32 seconds whereas GoogLeNet took an average of 16 minutes 27 seconds to train a model with the same amount of training and validation data. For testing, DFUNet took an average of 49 seconds whereas GoogLeNet took an average of 72 seconds to classify the same test data. Therefore, we demonstrated how reducing the number of layers using the bespoke architecture of DFUNet markedly reduced processing time, while also achieving higher sensitivity and specificity with the introduction of blocks of convolutional layers in parallel with an increased number of filter input.

Our proposed DFUNet received highest scores in performance measures in *Sensitivity*, with a score of 0.934, *F-measure* with 0.939 and *AUC* with 0.962. Whereas in *Specificity* and *Precision*, the scores achieved in these performance measures by DFUNet and GoogLeNet are almost the same.

With data augmentation technique, these patches were made 15 times for both training and validation. But, when we tested the data augmentation training in our experiment, there were no differences found in performance metrics with all the models. Hence, we did not include the data augmentation results in Table 5.3 and Table 5.4 as it did not improve the results. The main reasons behind the failure of data augmentation were overall performance metrics recorded without data augmentation was quite high and there was only small number of misclassification cases which were not corrected even with models trained with data augmentation. For example, we found in evaluations, some ulcer conditions with similar skin tone and small sizes shown in the second row in Fig. 5.9 have too subtle features to be detected as ulcer regardless of any pre-processing with data-augmentation. Hence, we did not use data augmentation to produce final results as the training with data augmentation become more computational expensive because of 15 times more data than the normal dataset. Also, the focus of this work is to determine the skin lesions are at high risk to be detected as misclassification.

There is no evidence of an influence of factors such as lighting conditions and skin tone due to patient's ethnicity on DFU classification. As ulcer and surrounding skin has quite distinctive texture and color features from the normal healthy skin irrespective of above-mentioned factors. In our experiments, these factors result in very few misclassified instances in testing set when there is very high red skin tone as shown in Fig. 5.9.

5.3.1 Experimental Analysis and Discussion

Diagnosis and detection of DFU by the computerized method has been an emerging research area with the evolution of computer vision, especially deep learning methods. This preliminary experiment of binary classification of DFU and healthy skin is performed to learn the distinctive features of both types of skin lesions. Also, the main motivation of this experiment to find the type of skin lesions which are at high risk of being misclassified by algorithms. In this experiment, we proposed a new lightweight deep learning architecture which can classify DFU and healthy skin lesions with high accuracy. There are a few examples of correctly and incorrectly classified cases in both abnormal and normal classes by DFUNet as illustrated in Fig. 5.9. The computer vision algorithms struggle to classify the very subtle DFU with similar skin tone correctly. They are detected as normal

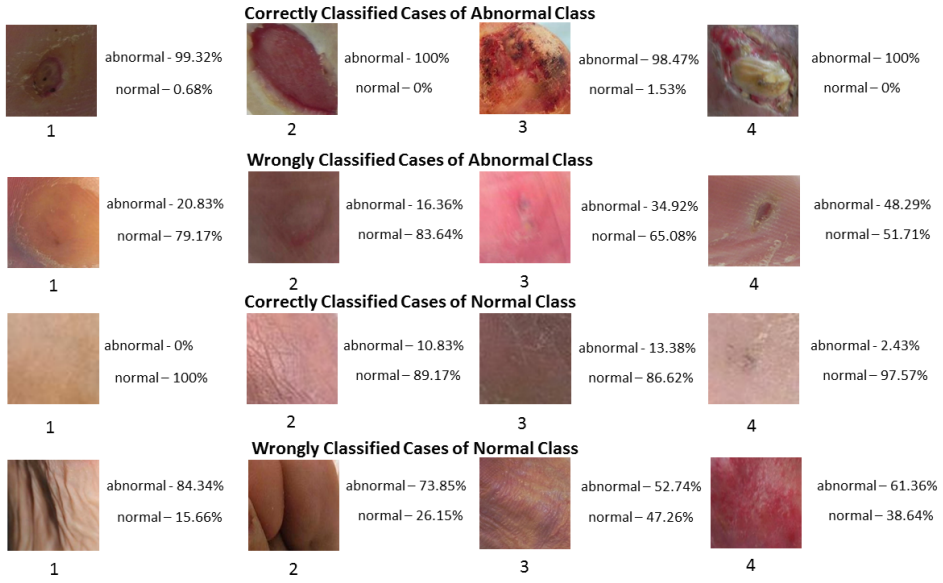


FIGURE 5.9: Few examples of accurate and inaccurate classified cases for both abnormal and normal classes with DFUNet.

with high percentage as illustrated by example 1 and 2 of misclassification cases of abnormal class in Fig. 5.9. Also, DFU that are very small in size is misclassified as normal as shown by example 3 and 4 of misclassification cases of abnormal class Fig. 5.9. In normal skin, the patches with toe, highly wrinkled skin, and very high red tone skin are classified wrongly by the proposed method as illustrated by the examples of misclassified cases of normal classes in Fig. 5.9.

5.4 Performance evaluation on Heterogeneous Test Case

Since, DFU dataset is captured with the same DSLR camera as mentioned in above section. With computer vision techniques, it is preferable to have heterogeneous capture to form dataset. But, strict medical ethical approval does not allow to use different cameras to capture the pictures of DFU in the healthcare setting. Hence, we collected another heterogeneous dataset of standardised DFU images with the help of FootSnap application. These images are captured with the help of IPad camera. We tested our algorithm on this heterogeneous dataset and received good performance with *Sensitivity* score of 0.929, *F-measure* with 0.931, *Specificity* of 0.908, *Precision* with 0.942 and *AUC* with 0.950 score.

Normal Spot Wrinkle

FIGURE 5.10: The examples of three classes in facial skin dataset.

TABLE 5.5: Facial Skin classification task with three classes as Normal skin, Spot, Wrinkle. The proposed DFUNet outperformed GoogLeNet in every performance metrics on this dataset.

	Sensitivity	*Specificity*	*Precision*	*Accuracy*	*F-Measure*	*MCC*
LBP	0.733	0.740	0.740	0.808	0.735	0.586
LBP + HOG	0.736	0.742	0.742	0.811	0.738	0.591
LBP + HOG + Colour	0.741	0.741	0.742	0.815	0.741	0.597
GoogLenet	0.783	0.882	0.784	0.846	0.784	0.665
Proposed DFUNet	**0.867**	**0.930**	**0.867**	**0.907**	**0.867**	**0.796**

5.5 Performance Evaluation on Facial Skin Dataset

Since, DFUNet performed well on the classification of DFU skin patches, to test the robustness of DFUNet on other skin lesion datasets, we run the experiment of 3-class classification of facial skin patches i.e. normal, spot and wrinkles as shown in the Fig. 5.10. It is worth mentioning, there is no public skin lesion dataset available for research without prior written consent. In this derma dataset, we delineated the equal number of skin patches i.e. 110 patches for each class. We used traditional machine learning methods and two best performing CNN architectures in Table. 5.4 i.e. GoogLeNet and DFUNet for this experiment. With the same experimental settings, DFUNet outperforms GoogLeNet and other methods in each evaluation metrics for 10-fold cross-validation data as shown in Table 5.5. This is due to the deep learning models are generally trained with datasets of a substantial amount of images to achieve good accuracy [122]. But, larger filter sizes in the later blocks of convolutional layers in parallel in DFUNet to extract more multiple features as compared to GoogLeNet improved the performance of DFUNet on this smaller dataset of 330 images.

5.6 Summary

In this work, we trained various classifiers based on traditional machine learning algorithms, CNNs and proposed a new CNN architecture, DFUNet on DFU classification which discriminates the DFU skin from healthy skin. With high-performance measures in classification, DFUNet allows the accurate automated detection of DFU in foot images and make it an innovative technique for DFU evaluation and medical treatment. For the detection of DFU, it is vital to understand the difference between DFU and healthy skin to know the features differences between these two classes in computer vision perspective. For classification, DFUNet is a light-weight CNN framework that is used for DFU dataset consists of two classes (ulcer and normal skin) and facial skin dataset consists of three classes (spot, wrinkles and normal skin), it will be further tested in the future to include many more classes. Therefore, we demonstrated how reducing the number of layers and number of neurons in FC layers using the bespoke architecture of DFUNet markedly reduced processing time, while also achieving higher sensitivity and specificity.

Chapter 6

DFU Segmentation

This Chapter presents the use of fully convolutional networks for automatic segmentation of DFU and its surrounding skin. Using 5-fold cross-validation, the proposed two-tier transfer learning FCN Model achieved a Dice Similarity Coefficient of 0.794 (±0.104) for ulcer region, 0.851 (±0.148) for surrounding skin region, and 0.899 (±0.072) for the combination of both regions.

6.1 Introduction

This work investigated a two-tier transfer learning from bigger datasets to train the FCNs to automatically segment the DFU and surrounding skin. Since even for specialist podiatrists, it is very hard to define the difference in the boundary between the DFU and it's surrounding skin. It is mainly because of high intra-class and inter-class visual similarities in tissues between these two classes and irregular contours. Hence it is a very difficult problem for any computer vision algorithm to clearly define and segment these two classes in the same region. This experiment was performed to evaluate the performances of deep learning algorithms to segment DFU and it's surrounding skin separately. The contributions of this work include

1. To overcome the deficiency of DFU dataset in the state of the art, we presented the largest DFU dataset and annotated ground truth (600 foot images with DFU).

TABLE 6.1: Segmentation results for color segmentation and traditional machine learning

Method	*Dice Similarity Coefficient*	*Specificity*	*Sensitivity*	*MCC*
	Complete	Complete	Complete	Complete
Color Segmentation	0.415	0.922	0.511	0.394
Traditional Machine Learning	0.533	0.938	0.575	0.507

2. This is the first attempt in computer vision methods to segment the significant surrounding skin separately from the ulcer.

3. We proposed a two-tier transfer learning method by training the FCNs on larger datasets of images and use it as a pre-trained model for the segmentation of ulcers and its surrounding skin. The performance was compared to other deep learning framework and the state-of-the-art ulcer/wound segmentation algorithms on our dataset.

The skin surrounding an ulcer is very important as its condition determines if the ulcer is healing and is also a vulnerable area for extension [123, 124]. There are many factors that increase the risk of vulnerable skin such as ischemia, inflammation, abnormal pressure, maceration from exudates etc. Similarly, healthy skin around the ulcer indicates a good healing process. Surrounding skin is examined by inspection of color, discharge and texture, and palpation for warmth, swelling and tenderness. On visual inspection, redness is suggestive of inflammation, which is usually due to wound infection. The black discoloration is suggestive of ischemia. White and soggy appearance is due to maceration and white and dry is usually due to increased pressure. It is important to recognise that skin appearances look different in different shades of skin. Lesions that appear red or brown in white skin, may appear black or purple in black or brown skin. Mild degrees of redness may be masked completely in dark skin.

6.2 Methodology

This section describes semantic segmentation of DFU and surrounding skin using Traditional Machine Learning (TML) methods and deep learning methods. In the end of this section, the performance metrics are used to compare FCN.

6.2.1 Traditional Machine Learning Methods for DFU Segmentation

In this section, we assessed the performance of TML methods for segmentation of DFU in DFU dataset. As these methods are not meant for surrounding skin segmentation, we re-implemented the state of the art on general ulcer/wound segmentation (henceforth complete). In image processing, we used the color segmentation with the different threshold value to get the desirable results of segmentation. For traditional machine learning, we delineated approximately 1780 patches (including 743 normal skin patches and 1037 abnormal skin patches) for feature extraction and training of classifier using 5-fold validation from the 480 original foot images. Since, these two classes of skin (normal and abnormal) have major textural differences amongst them, we investigated the popular feature extraction techniques including texture descriptors such as LBP [87],HOG [89] and color descriptors such as Normalised *RGB, HSV, and L*u*v* features [91]. After the feature extraction from images, we used Quadratic support vector machine [125] as classifier for classification task. Then, to perform segmentation task, we used the sliding window approach to mask each pixel if the corresponding patch is detected as ulcer by trained classifier.

Both techniques have achieved a very average score in evaluation metrics, such as color segmentation achieved 0.415 ($\pm$0.208) in *Dice Similarity Coefficient* and traditional machine learning is slightly better and achieved 0.533 ($\pm$0.223) in *Dice Similarity Coefficient*. The complete evaluation for both techniques on the testing dataset is illustrated by Table 6.1. These conventional segmentation methods require a lot of intermediate steps like pre-processing of images, extracting hand-crafted features and rigorous manual-tuning of parameters to get the results. Whereas, deep learning provides the end-to-end models on various computing platforms which simply take images as input and provide the final segmentation results as output.

6.2.2 Fully Convolutional Networks for DFU segmentation

Deep learning models proved to be powerful algorithms to retrieve hierarchies of features to achieve various tasks of computer vision. These convolutional neural networks, especially classification networks have been used to classify various

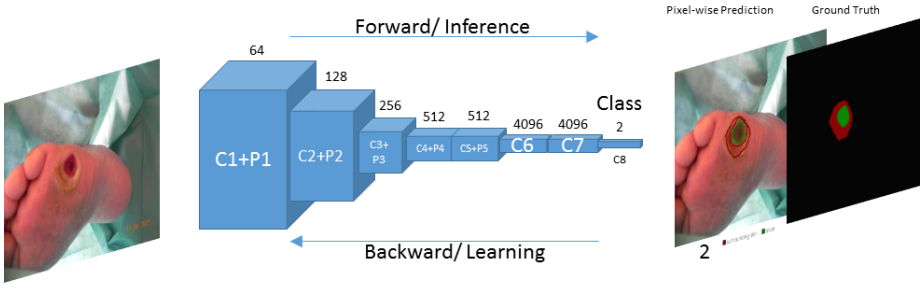


FIGURE 6.1: Overwiew of fully convolutional network's architecture which can learn features with forward and backward learning to make pixel-wise prediction to perform segmentation where C1-C8 are convolutional layers and P1-P5 are max-pooling layers

classes of objects by assigning discrete probability distribution for each class. But, these networks have limitations as they are not able to classify multiple classes in a single image and figure out the position of the objects in images. FCNs instead produce segmentation by addressing these limitations by pixel-wise prediction rather than single probability distribution in the classification task for each image. Therefore, each pixel of an image is predicted for which class it belongs. The working of FCN architecture to produce pixel-wise prediction with the help of supervised pre-training using the ground truth is illustrated in Fig. 6.1. Hence, these models have the ability to predict multiple objects of various classes and the position of each object in images.

6.2.2.1 FCN-AlexNet

The FCN-AlexNet is a fully convolutional network version of original classification model AlexNet by few adjustments of layers of networks for segmentation [126]. This network was originally used for classification of 1000 different objects of classes on the ImageNet dataset. It emerged as winner of imageNet ILSVRC-2012 competition in classification category by achieving 99% confidence [3]. There are few customisations made in the classification network model in order to convert it into FCN to carry out dense prediction. In FCN-AlexNet, earlier CNN layers are kept the same to extract the features and fully connected layers which throw away the positional coordinates are convolutionalised with the equivalent convolutional layers by adjusting the size of filters according to the size of the input to these layers [126]. After the extraction of coarser and high-level features from input images, to produce the pixel-wise prediction for every pixel of the input, the deconvolutional

layers work exactly opposite to the convolutional layers and stride used in this layer is equal to the scaling factor used in the convolutional layers.

The input was 500×500 foot images and ground truth images (Pascal VOC format). In the end, the network prediction on test images was very close to the ground truth. We used the Caffe [116] framework to implement FCN-AlexNet. We used these network parameters to train a model on the dataset i.e. 60 epochs, a learning method as stochastic gradient descent as rate of 0.0001 with a step-down policy and step size of 33%, and gamma is 0.1. The learning parameter is decreased by the factor of 100 due to the introduction of new convolutional layers instead of fully connected layers which result in improved performance of FCN-AlexNet and other FCNs.

6.2.2.2 FCN-32s, FCN-16s, FCN-8s

FCN-32s, FCN-16s, and FCN-8s are three models inspired by the VGG-16 based net which is a 16 layer CNN architecture that participated in the ImageNet Challenge 2014 and secured the first position in localisation and second place in classification competition [126, 127]. These models are customised with the different upsampling layers that magnify the output used in the original CNN model VGG-16. FCN-32s is same as of FCN-VGG16 in which fully connected layers are convolutionised and end to end deconvolution is performed with 32-pixel stride. The FCN-16s and FCN-8s additionally work on low-level features in order to produce more accurate segmentation. In FCN-16s, the final output is the sum of upsampling of two layers i.e. upsampling of pool4 and 2× upsampling of convolutional layer 7 whereas in FCN-8s, it is the sum of upsampling of pool3, 2× upsampling of pool4 and 4× upsampling of convolutional layer 7. Both models perform prediction on much more finer grained analysis i.e. 16×16 pixel blocks for FCN-16s and 8×8 pixel blocks for FCN-8s. The suitable pre-trained models for each model are also used in the training. The same input images are used to train the model with the same parameters as of FCN-AlexNet i.e. 60 epochs, a learning rate of 0.0001, and gamma of 0.1.

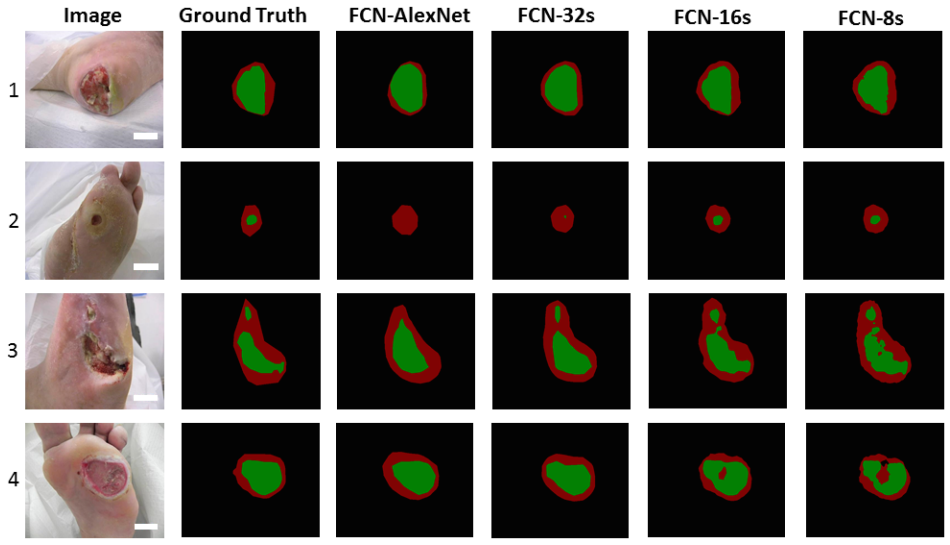


FIGURE 6.2: Four Examples of DFU and surrounding skin segmentation with the help of four different FCN models

6.3 Experiment and Result

As mentioned previously, we used the deep learning models for the segmentation task. The experiments were carried out on the DFU dataset of 600 ulcer foot images that were split into the 70% training, 10% validation and 20% testing. We adopted a 5-fold cross-validation. For training and validation using the deep learning architecture, we used 420 images and 60 images respectively from the 600 original foot ulcer images. Finally, we tested our model predictions on 120 remaining images. Further, we tested the performance of the models on 105 healthy test images.

The performance evaluation of the FCN frameworks on the testing set is achieved with 3 different DFU regions due to the practical medical applications. The DFU regions are explained below:

1. The complete area determination (including Ulcer and Surrounding Skin).
2. The ulcer region
3. The surrounding skin (SS) region

In Table 5.4, we report *Dice Similarity Coefficient (Dice)*, *Sensitivity*, *Specificity*, *Mathews Correlation Coefficient (MCC)* as our evaluation metrics for segmentation of DFU region. In medical imaging, *Sensitivity* and *Specificity* are

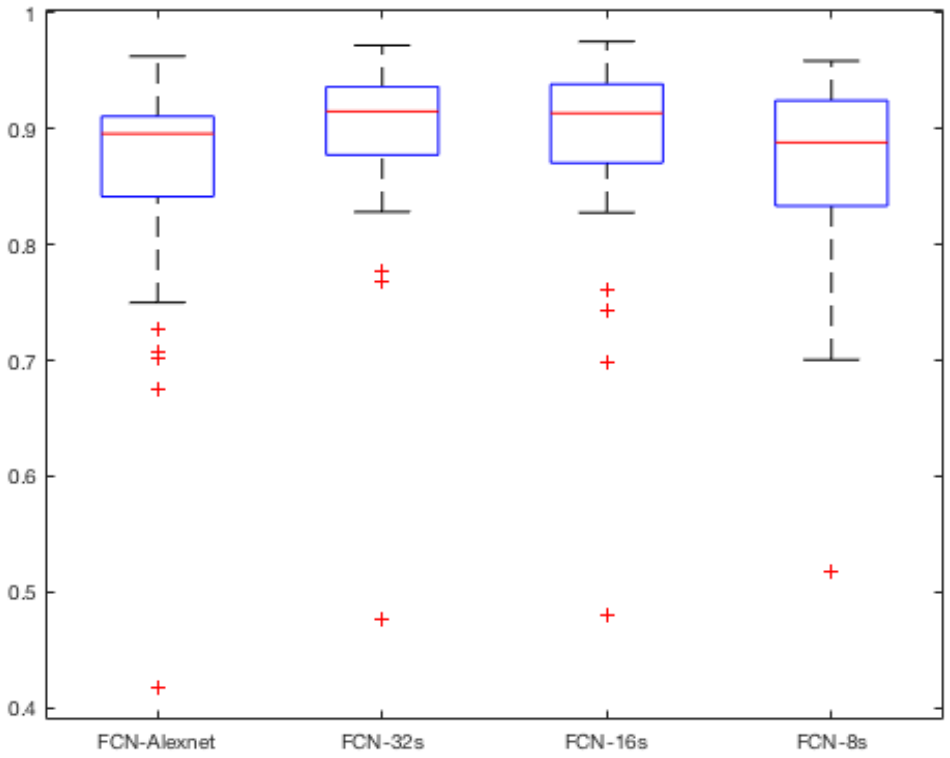


FIGURE 6.3: Boxplot of *Dice* for all FCN models for Complete Area Determination

TABLE 6.2: Comparison of different FCNs architectures on DFU dataset (SS denotes Surrounding Skin)

Method	*Dice*			*Specificity*			*Sensitivity*			*MCC*		
	Complete	Ulcer	SS	Complete	Ulcer	SS	Complete	Ulcer	SS	Complete	Ulcer	SS
FCN-AlexNet	0.869	0.707	0.685	0.985	0.982	0.991	0.879	0.714	0.731	0.859	0.697	0.694
FCN-32s	**0.899**	0.763	0.762	0.989	0.986	0.989	**0.904**	0.751	0.823	**0.891**	0.752	0.768
FCN-16s	0.897	**0.794**	**0.851**	0.988	0.986	**0.994**	0.900	**0.789**	**0.874**	0.889	**0.785**	**0.852**
FCN-8s	0.873	0.753	0.835	**0.990**	**0.987**	0.993	0.854	0.726	0.847	0.865	0.744	0.838

considered reliable evaluation metrics and where as for segmentation evalaution, *Dice* are popularily used by researchers.

In performance measures, FCN-16s was the best performer and FCN-AlexNet emerged as the worst performer for various evaluation metrics among all the other FCN architectures. Though, FCN architectures achieve comparable results when the evaluation is considered in the complete region. But, there is a notable difference in the performance of FCN models when ulcer and especially surrounding skin regions are considered. FCN-16s has achieved the best score of 0.794 ($\pm$0.104) in the ulcer region and 0.851 ($\pm$0.148) in the surrounding skin region for *Dice*. whereas the FCN-32s achieved the best score of 0.899 ($\pm$0.072) in the complete area determination. The boxplots for all the FCN models performance in *Dice*

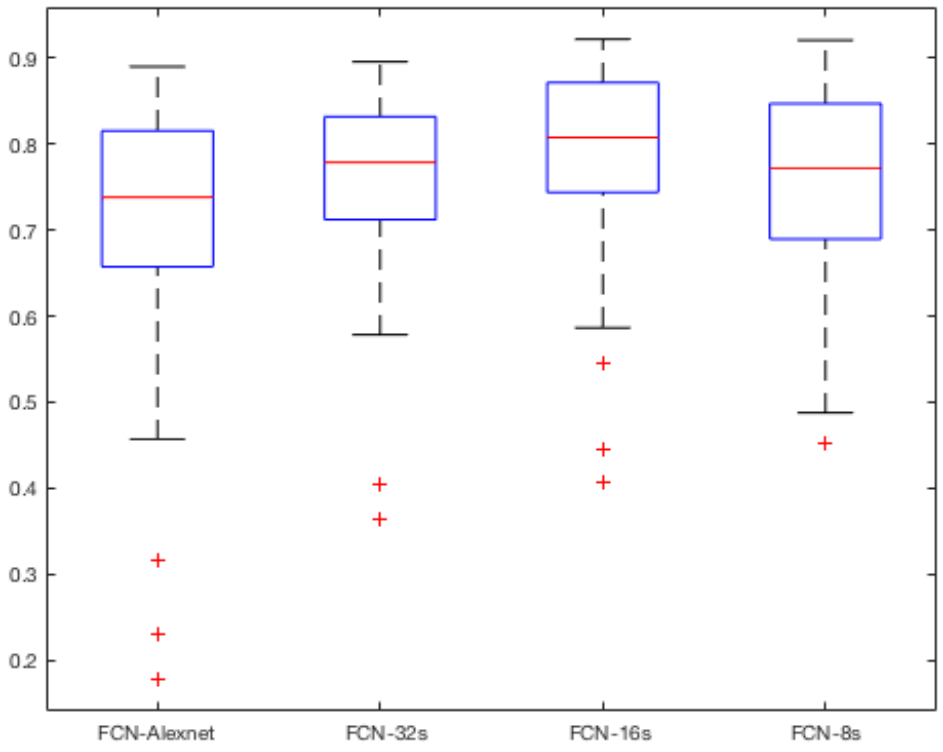


FIGURE 6.4: Boxplot of *Dice* for all FCN models for Ulcer region

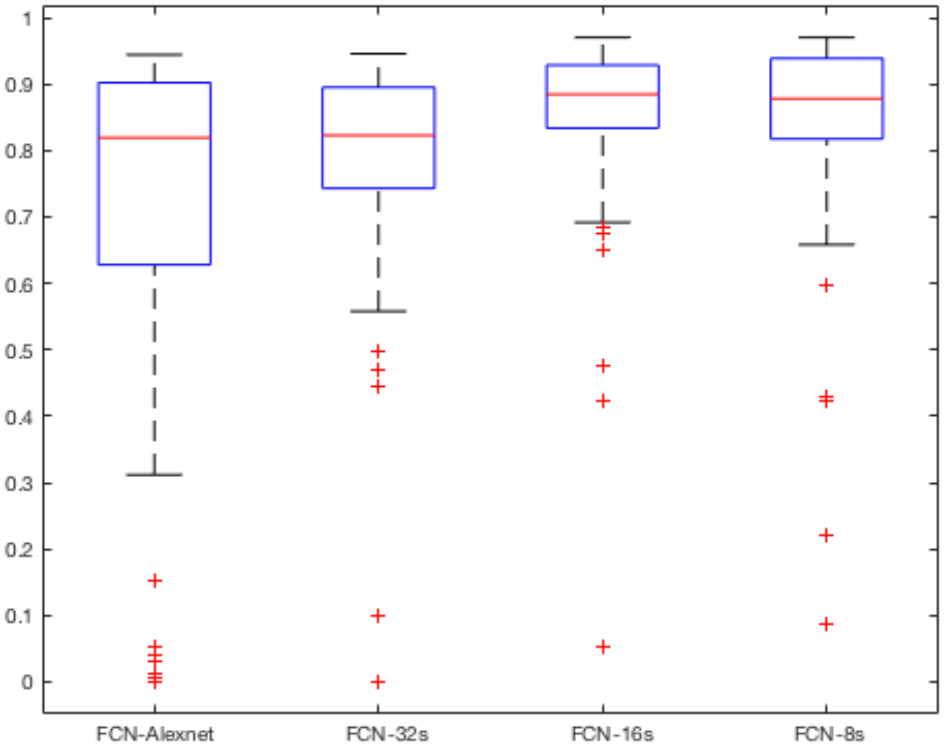


FIGURE 6.5: Boxplot of *Dice* for all FCN models for Surrounding Skin region

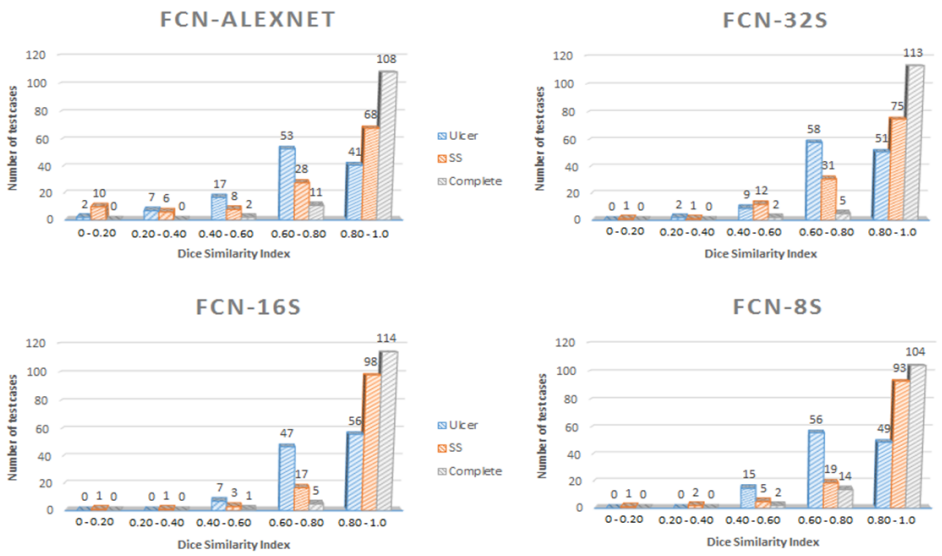


FIGURE 6.6: Distribution of Dice Similarity Coefficient for each trained model

separately for all three regions are illustrated through Fig. 6.3, Fig. 6.4 and Fig. 6.5. Overall, the FCN models has very high *Specificity* for all the regions. Further, assessing the FCNs performance, we observed that FCN-16s and FCN-32s are better in *Senstivity*. FCN-16s performed best in the ulcer and surrounding skin regions and FCN-32s has the best in complete region performance in segmenting the complete region in terms of *Sensitivity*, *Dice* and *MCC*. The results in Table 6.2 showed that the complete region segmentation has better performance than ulcer and surrounding skin in terms of Dice and MCC.

Finally, we tested the performance of the trained models on healthy foot images, they produced the highest specificity of 1.0 where neither ulcer nor surrounding skin was detected.

6.3.1 Inaccurate segmentation cases in FCN-AlexNet, FCN-32s, FCN-16s, FCN-8s

Although the results are promising, there are few inaccurate segmentation cases that achieve very *Dice* for each trained model as shown in Fig. 6.6. The examples of such cases for every FCNs that we trained are illustrated in the Fig. 6.7. There are few instances in which FCN-AlexNet and FCN-32s models are not able to detect the small ulcers and distinct surrounding skin or detect a very small part of them. As discussed earlier, ulcer and surrounding skin regions have very irregular outer boundaries, FCN-AlexNet and FCN-32s always tend to draw more

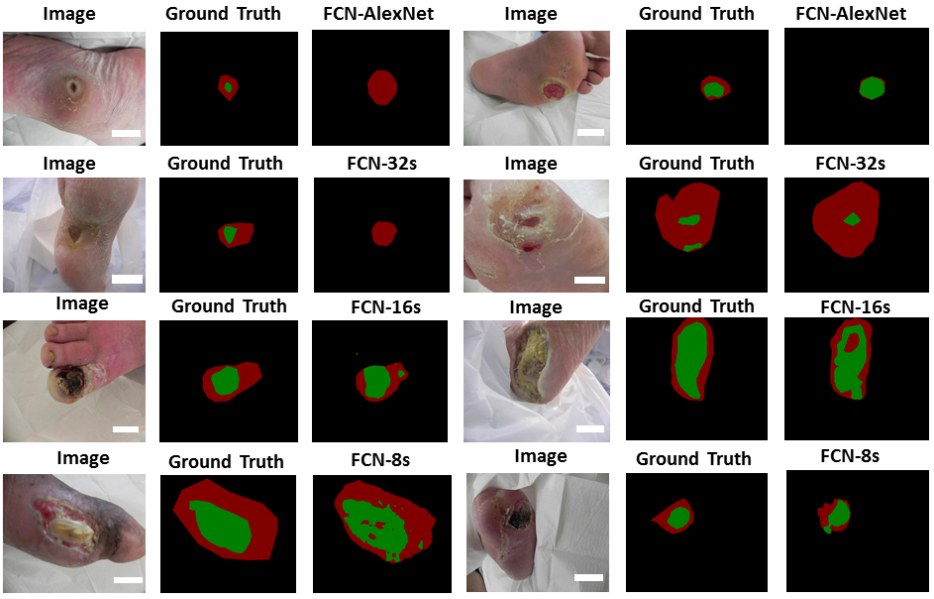


FIGURE 6.7: Inaccurate segmentation cases by the different FCNs used in the testing dataset

regular contour and struggled to draw irregular boundaries to perform accurate segmentation, whereas, FCN-16s and FCN-8s with smaller pixel stride were able to produce more irregular contours of both ulcer and surrounding skin. But, in a few test images, some part of both categories overlap in some region due to the distinct tissues of ulcer looks like surrounding skin and vice versa.

6.4 Summary

In this work, we developed and applied computer vision and deep learning approaches to train various FCNs that can automatically detect and segment the DFU and surrounding skin area with a high degree of accuracy. It's important to segment the surrounding skin along with DFU as surrounding skin is an important hint of the progress of DFU. This work also lays the foundations for technology that may transform the preliminary examination of diabetic foot ulcers. Moreover, this research could be applied to other related medical fields, for example, in automatically identifying and segmenting a range of other skin lesions from images of the pathologies.

Chapter 7

DFU Localisation

This Chapter presents robust deep learning methods for DFU localisation on foot images. We demonstrated the application of this work by transferring a lightweight DFU localisation model to mobile devices for remote monitoring of DFU.

7.1 Introduction

Deep learning methods for object localisation task in the computer vision and medical imaging field is drawing lots of attention of both researchers and developers. In the last few years, the accuracy of algorithms on public object localisation datasets is significantly improved with the introduction of CNNs. In this work, we provided a large-scale annotated DFU dataset, tested new and lightweight deep learning architectures such as Faster R-CNN, SSD, R-FCN on this DFU dataset of 1775 images and propose an end-to-end mobile solution for DFU localisation. The key contributions of this work include:

1. We presented one of the largest DFU dataset, which consists of 1775 images with annotated bounding box indicating the ground truth of DFU location. To date, the largest dataset we encountered is of 600 DFU images, where it was used for the semantic segmentation of DFU and its surrounding skin [25].

2. We proposed the use of CNNs to localise DFU in real-time with two-tier transfer learning. To our best knowledge, this is the first time CNNs are used for this task. Since our main focus is on mobile devices, we emphasised on light-weight object localisation models.

3. Finally, we demonstrated the application of our proposed methods on two types of mobile devices: Nvidia Jetson TX2 and an android mobile application.

The major challenges of DFU localisation task are as follow: 1) Expensive in data collection and expert labelling on the DFU dataset; 2) High inter-class similarity between the DFU lesions and intraclass variation depending upon the classification of DFU [1]; and 3) Lighting conditions and patient's ethnicity.

7.2 Methodology

This section describes a brief description of deep learning methods for DFU localisation. We compared these methods with popular localization performance metrics.

7.2.1 Traditional Methods for DFU Localisation and Classification

In this section, we assessed the performance of conventional methods for the localisation of DFU. For traditional machine learning, we annotated 2028 normal skin patches and 2080 abnormal skin patches from expert annotations for DFU localisation. We utilized this dataset for feature extraction and training of classifier using 5-fold cross-validation [26]. We also used data-augmentation techniques such as flipping, rotation, random crop, color channels to make a total of 28392 normal and 29120 abnormal patches. 80% of the image data is used to train the classifier and remaining 20% of the data is used as test images. Since these two classes of skin (normal and abnormal) have significant textural and color differences amongst them, we utilized LBP, HOG, color descriptors to extract features from skin patches of both normal and abnormal classes. For a single patch, 209

features were extracted with above-mentioned feature extraction techniques. After the feature extraction from images, we used support vector machine [125] as a classifier for the classification task. Then, to perform DFU localisation task with multiple scales, we used the sliding window approach to mask each box if the corresponding patch is detected as ulcer by a trained classifier.

This technique has achieved a good score in evaluation metrics, 70.3% in *Mean Average Precision*. The traditional machine learning methods require a lot of intermediate steps like pre-processing of images, extracting hand-crafted features and multiple stages to get the final results which makes them very slow. Whereas, deep learning provides the faster end-to-end models on various computing platforms which simply take images as input and provide the final localisation results as output.

7.2.2 Deep Learning Methods for DFU Localisation

CNNs proved their superiority compared to the conventional machine learning techniques in image recognition tasks such as ImageNet [3] and MS-COCO challenges [5]. They are very capable of classifying the images into different classes of objects from both non-medical and medical imaging by extracting the hierarchies of features. One of the important tasks in computer vision is object localisation where algorithms need to localise and identify the multiple objects in an image. Mainly, object localisation networks consist of three stages as described in the following subsections.

7.2.2.1 CNN as feature extractor

In Stage 1, the standard CNN such as MobileNet, InceptionV2, the convolutional layers extract the features from input images as feature maps. These feature maps are used to identify the objects in the image with particular attention focused on DFU regions as shown in Fig. 7.1. These feature maps serve as input for the later stages such as the generation of proposals in the second stage and classification and regression of RoI in the third stage.

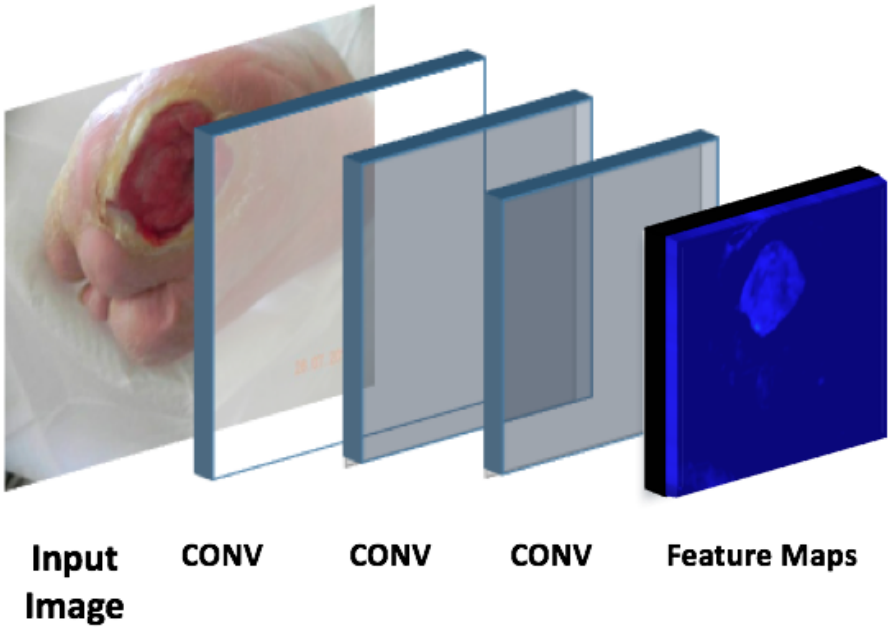


FIGURE 7.1: Stage 1: The feature map extracted by CNN that acts as backbone for object localisation network. Conv refers convolutional layer.

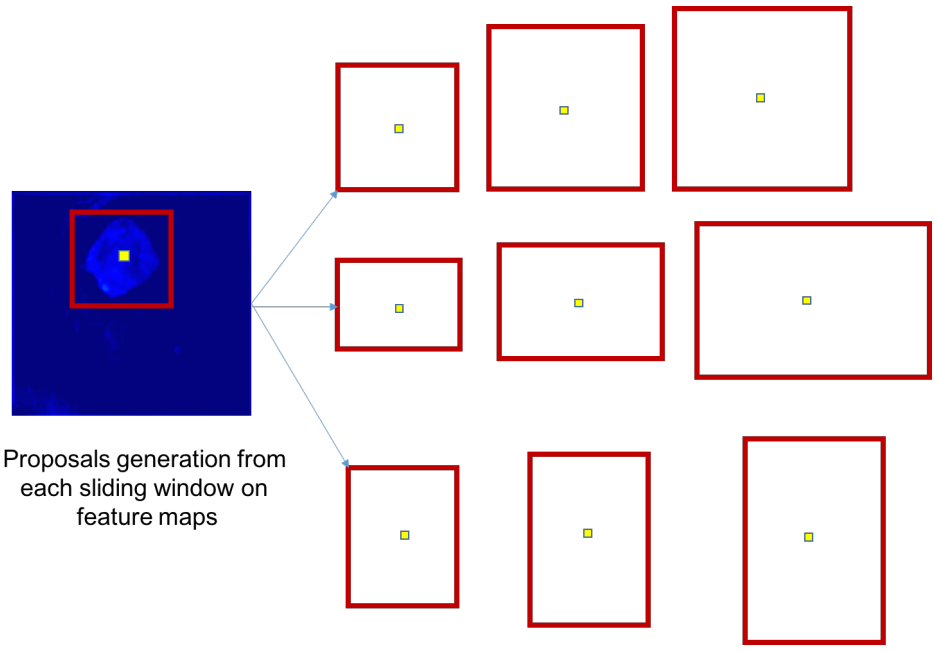


FIGURE 7.2: Stage 2: Detected proposal boxes with translate/scale operation to fit the object. There can be several proposals on a single object.

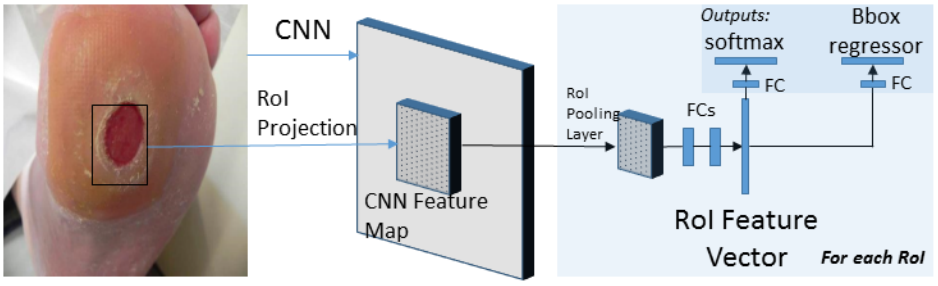


FIGURE 7.3: Illustration of Stage 3: The classification and further box refinement of RoI boxes from the second stage proposal with softmax and Bbox regression. Where FC refers to Fully-connected layer

7.2.2.2 Generation of proposals and refinement

In Stage 2, the network scans the image in a sliding-window fashion and finds specific areas that contain the objects using the feature map extracted in Stage 1. These areas are known as proposals which have different boxes distributed over the image. In general, around 200,000 proposals of different sizes and aspect ratios are found to cover as many objects as possible in the image. With GPU, Faster R-CNN produces these much anchors in 10ms [128]. Stage 2 generates two outputs for each proposal:

- Proposal Class: It can be either foreground or background. The foreground class means there is likely an object in that proposal and it is also known as a positive proposal.
- Proposal Refinement: A positive proposal might not be perfectly capture the object. So the network estimates a delta (% change in x, y, width, height) for refinement of the proposal box to centre the object better as illustrated in Fig. 7.2.

7.2.2.3 RoI Classifier and Bounding Box Regressor

Stage 3 consists of the classification of RoI boxes provided by Stage 2 and further refinement of the RoI boxes as shown in Fig. 7.3. First, all RoI boxes are fed into the RoI pooling layer to resize them into fixed input size for classifier as RoI boxes can have different sizes. Similar to Stage 2, it generates two outputs for each RoI:

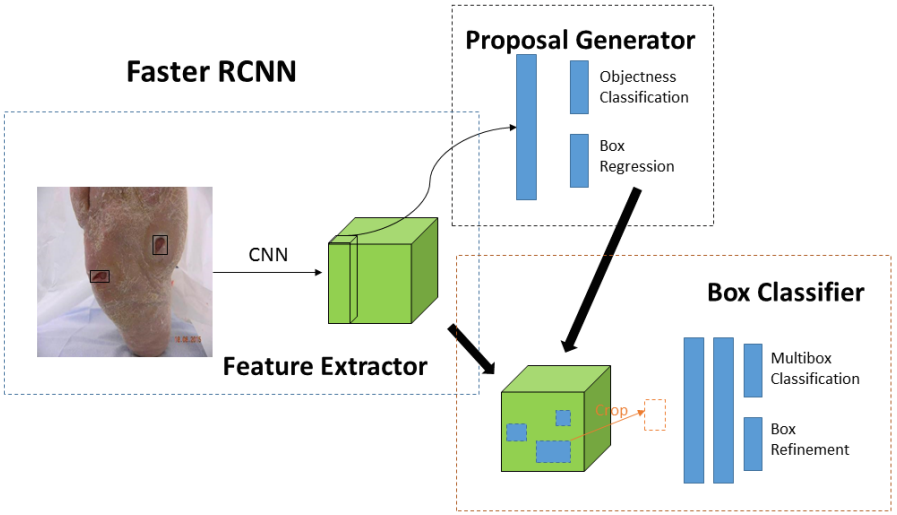


FIGURE 7.4: Faster R-CNN Architecture for DFU localisation which consists of all three stages discussed earlier.

- RoI Class: The softmax layer provides the classification of regions to specific classes (if more than one class). If the RoI is classified as a background class, it is discarded.
- Bbox Refinement: Its purpose is to refine the location of RoI boxes.

We considered three types of object localisation networks to perform on the DFU dataset. First is Faster R-CNN [128], which is a successor of Fast R-CNN [129] for object localisation in terms of speed. It consists of all three stages of object localisation network as shown in Fig. 7.4. It has two-stage loss function whereas first stage loss function that consists of the parameters such as space, scale and aspect ratio of the proposals. Then, second stage loss function re-runs the crops of proposal produced by the second stage with feature extractor to produce more accurate box proposals for classification.

Dai et al. [130] proposed the Region-based Fully Convolutional Networks (R-FCN) to produce faster box proposals by considering the crops only from the last layer of features with comparable accuracy as Faster R-CNN which crop features from the same layer where region proposals are predicted as shown in the Fig. 7.5. Due to cropping limited only to the last layer, it minimizes the time to get the box refinement.

Single Shot Multibox Detector (SSD) [131] is a new architecture for the object localisation which uses a single stage CNN to predict classes directly and anchor offsets without the need of second stage proposal generator unlike Faster R-CNN

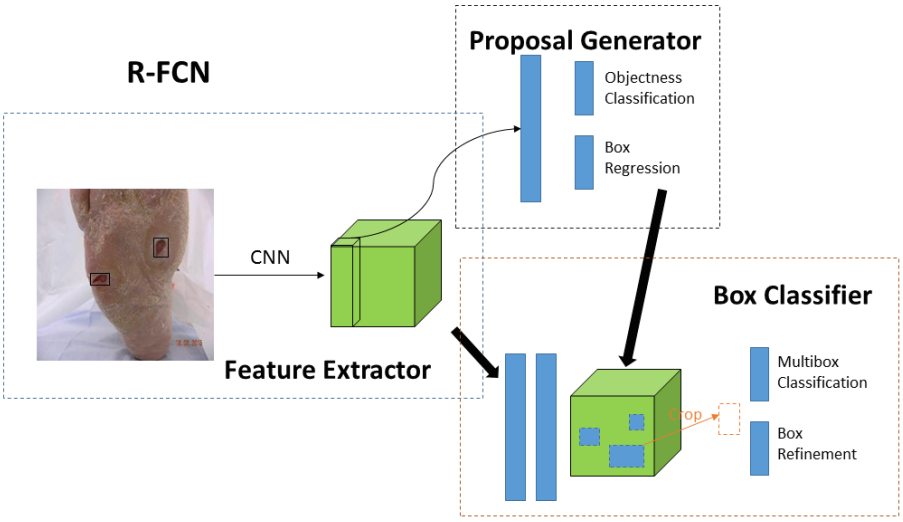


FIGURE 7.5: R-FCN Architecture which considers only the feature map from the last convolutional layer which speeds up the three stage network

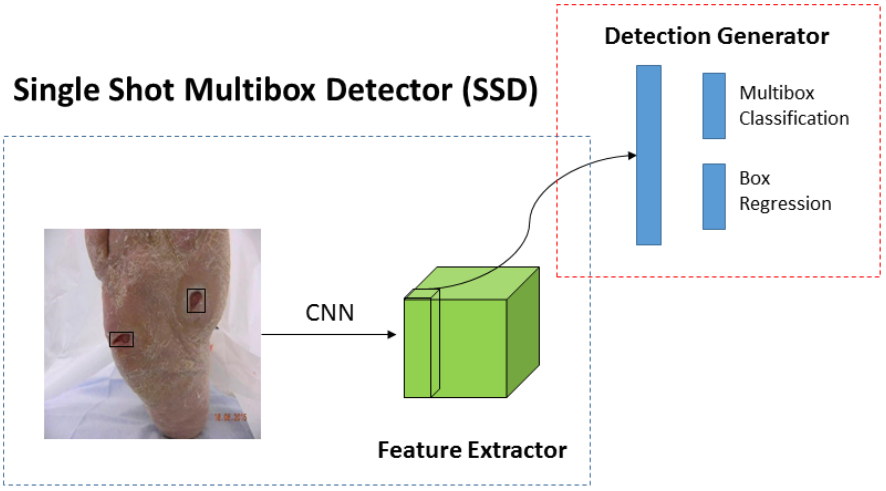


FIGURE 7.6: The architecture of Single Shot Multibox Detector (SSD). It considers only two stage by eliminating the last stage to produce faster box proposals.

[128] and R-FCN [130] as shown in the Fig. 7.6. The SSD meta-architecture produces anchors much faster than other object localisation networks, which makes it more suitable for the mobile platforms.

There are six popular state-of-the-art object localisation models which are based on these three region based detector meta-architectures i.e. Single Shot multibox detector [131], R-FCN [130] and Faster R-CNN [128]. These three meta-architectures used the state-of-the-art classification algorithms like MobileNet [132], InceptionV2 [133], ResNet101 [120], Inception-ResNetV2 [134] to get the anchor boxes from the features maps, and finally, classify these anchors to different classes.

TABLE 7.1: Performance of state-of-the-art object localisation models on MS-COCO dataset. [5]

Model Name	*Speed (ms)*	*Size of Model (MB)*	*COCO mAP*
SSD-MobileNet	30	29.2	21
SSD-InceptionV2	42	102.2	24
Faster R-CNN with InceptionV2	57.2	58	28
R-FCN with ResNet101	92	218.3	30
Faster R-CNN with ResNet101	106	196.9	32
Faster R-CNN with Inception-ResnetV2	620	247.5	37

Table 7.1 summarises the size of models, speed (inference per image), and accuracy (mAP) trained on MS-COCO dataset with 90 classes [5, 135].

Since our work is limited by the hardware on mobile devices and real-time prediction, we only considered lightweight models (very small, low latency) in terms of the size of the model and inference speed. We used the first three models (SSD-MobileNet, SSD-InceptionV2 and Faster R-CNN with InceptionV2) for the DFU dataset as illustrated in Table 7.1. These small models are specifically chosen to match the resource restrictions (latency, size) on mobile devices for this application. To evaluate the performance of DFU localisation using a heavy model, we also included R-FCN with ResNet101 to our experiment.

Inception-V2 is a new iteration of the original inception architecture called GoogleNet with new features such as factorisation of bigger convolution kernels to multiple smaller convolution kernels and improved normalisation. For the first time, this network used depth-wise separable convolutions to reduce the computations in the first few layers. They also introduced batch normalisation layer which can decrease internal covariate shift, also combat the gradient vanishing problem to improve the convergence during training [133].

MobileNet is a recent lightweight CNN which uses depth-wise separable convolutions to build small, low latency models with a reasonable amount of accuracy that matches the limited resource on mobile devices. The basic block of depth-wise separable convolution consists of depth-wise convolution and pointwise convolution. The 3×3 depth-wise convolution is used to apply a single filter per each

input channel whereas pointwise convolution is just simple 1×1 convolution used to create the linear combination of the depth-wise convolution output. Also, it uses both batchnorm layers as well as RELU layers after both layers [132].

ResNet101 is one of the residual learning networks which won the first place on ILSVRC 2015 classification task [120]. As suggested by the name, ResNet101 is a very deep network consists of 101 layers which is about 5 times much deeper than VGG nets but still having lower complexity. The core idea of ResNet is providing shortcut connection between layers, which make it safe to train very deep network to gain maximal representation power without worrying about the degradation problem, i.e., learning difficulties introduced by deep layers.

7.2.3 Performance Measures of Deep Learning Methods

We used four performance metrics i.e. *Speed*, *Size of the model*, *mean average precision (mAP)*, and *Overlap Percentage*. The *Speed* determines the time model takes to perform inference on a single image whereas *Size of the model* is the total size of the frozen model that is used for the inference of test images. These are crucial factors for the real-time prediction on mobile platforms. The *mAP* has an "overlap criterion" of intersection-over-union greater than 0.5. The *mAP* is an important performance metric extensively used for the evaluation of the object localisation task. The prediction by a model to be considered a correct detection, the area of overlap A_o between the bounding box of prediction B_p and bounding box of ground truth B_g must exceed 0.5 (50%) [136]. The last evaluation metric is called *Overlap Percentage*, which is a mean average of intersection over union for all correct detection.

$$A_o = \frac{area(B_p \cap B_g)}{area(B_p \cup B_g)} \tag{7.1}$$

7.3 Experiment and Result

As mentioned previously, we used the deep learning models based on three meta-architectures for the DFU localisation task. Tensorflow object detection API [135] provides an open source framework which makes very convenient to design and

TABLE 7.2: Performance measures of object localisation models on DFU dataset

Model Name	*Speed (ms)*	*Size of Model (MB)*	*Ulcer mAP*	*Overlap Percentage (%)*
SSD-MobileNet	28	22.6	84.9	89.4
SSD-InceptionV2	37	53.5	87.2	92.6
Faster R-CNN with InceptionV2	48	52.2	91.8	95.8
R-FCN with Resnet 101	90	199.1	90.6	96.1

build various object localisation models. The experiments were carried out on the DFU dataset and evaluated with 5-fold cross-validation technique. First, we randomly split the whole dataset into 5 testing sets (20% each) for 5-fold cross-validation. This is to ensure that the whole dataset was evaluated on testing sets. For each testing set (20%), the remaining images were randomly split into 70% for the training set and 10% validation set. Hence, for each fold, we divided the whole dataset of 1775 images into approximately 1242 images in the training set, 178 in the validation set and 355 in the testing set. This was repeated for 5-fold to ensure the whole dataset was included in testing set.

Configuration of GPU Machine for Experiments (1) Hardware: CPU - Intel i7-6700 @ 4.00Ghz, GPU - NVIDIA TITAN X 12GB, RAM - 32GB DDR4 (2) Software: Tensor-flow [135].

We tested four state-of-the-art deep convolutional networks for our proposed object localisation task as described in Section III B. We trained the models with input-size of 640x640 using stochastic gradient descent with different learning rate on Nvidia GeForce GTX TITAN X card. We initialised the network with pre-trained weights using transfer learning rather than randomly initialised weights for the better convergence of the network. We tested the multiple learning rates by decreasing the original learning rates with the 10 and 100 times as well as multiplication factor from 1 to 5 to check the overall minimal validation loss. For example, if the original Inception-V2 learning rate was set at 0.001. Then, for training on DFU dataset, we used 10 learning rates of 0.0001, 0.0002, 0.0003, 0.0004, 0.0005, 0.00001, 0.00002, 0.00003, 0.00004, 0.00005.

We used 100 epochs for the training of each reported model, which we found are sufficient to train the DFU dataset as both training and validation loss finally converge to optimal lowest. We selected the models on the basis of minimum

validation losses for the evaluation. We tried different hyper-parameters such as learning rate, number of steps and data augmentation options for each model to minimize both training and validation losses. In the next section, we report the different network hyper-parameters and configurations for each model used for evaluation on the DFU dataset.

We set the appropriate hyper-parameters on the basis of meta-architecture to train the models on the DFU dataset. For SSD, we used two CNNs, MobileNet and Inception-V2 (both of them use depth-wise separable convolutions), we set the weight for `l2_regularizer` as 0.00004, initialiser that generates a truncated normal distribution with a standard deviation of 0.03 and mean of 0.0, `batch_norm` with the decay of 0.9997 and epsilon of 0.001. For training, we used a batch size of 24, optimizer as `RMS_Prop` with a learning rate of 0.004 and a decay factor of 0.95. The momentum optimizer value is set at 0.9 with a decay of 0.9 and epsilon of 0.1. We also used two types of data augmentation as random horizontal flip and random crop. For Faster R-CNN, we set the weight for `l2_regularizer` as 0.0, initialiser that generates a truncated normal distribution with standard deviation of 0.01, `batch_norm` with decay of 0.9997 and epsilon of 0.001. For training, we used a batch size of 2, optimizer as momentum with manual step learning rate with an initial rate as 0.0002, 0.00002 at epoch 40 and 0.000002 at epoch 60. The momentum optimizer value is set at 0.9. For training RFCN, we used the same hyper-parameters as Faster R-CNN with only change in the learning rate set as 0.0005. For data augmentation, we used only random horizontal flip for these two meta-architectures.

In Table 7.2, we report the performance evaluation of object localisation networks for DFU dataset on 5-fold cross-validation. Overall, all the models achieved promising localisation results with high confidence on DFU dataset. Few instances of accurate localisation by all trained models are demonstrated by Fig. 7.7. SSD-MobileNet ranked first in the *Size of Model* and *Average Speed* performance index. This is mainly due to the simpler architecture to generate anchor boxes in SSD [131]. Whereas in *Ulcer mAP* and *Overlap Percentage*, R-FCN with ResNet101 and Faster R-CNN with InceptionV2 were almost equally competitive in these performance measures. In *Ulcer mAP*, Faster R-CNN with InceptionV2 ranked first with an overall mAP of 91.8%, just slightly better than R-FCN with ResNet101 with mAP of 90.6%. But, in *Overlap Percentage*, R-FCN-Resnet101 achieved a

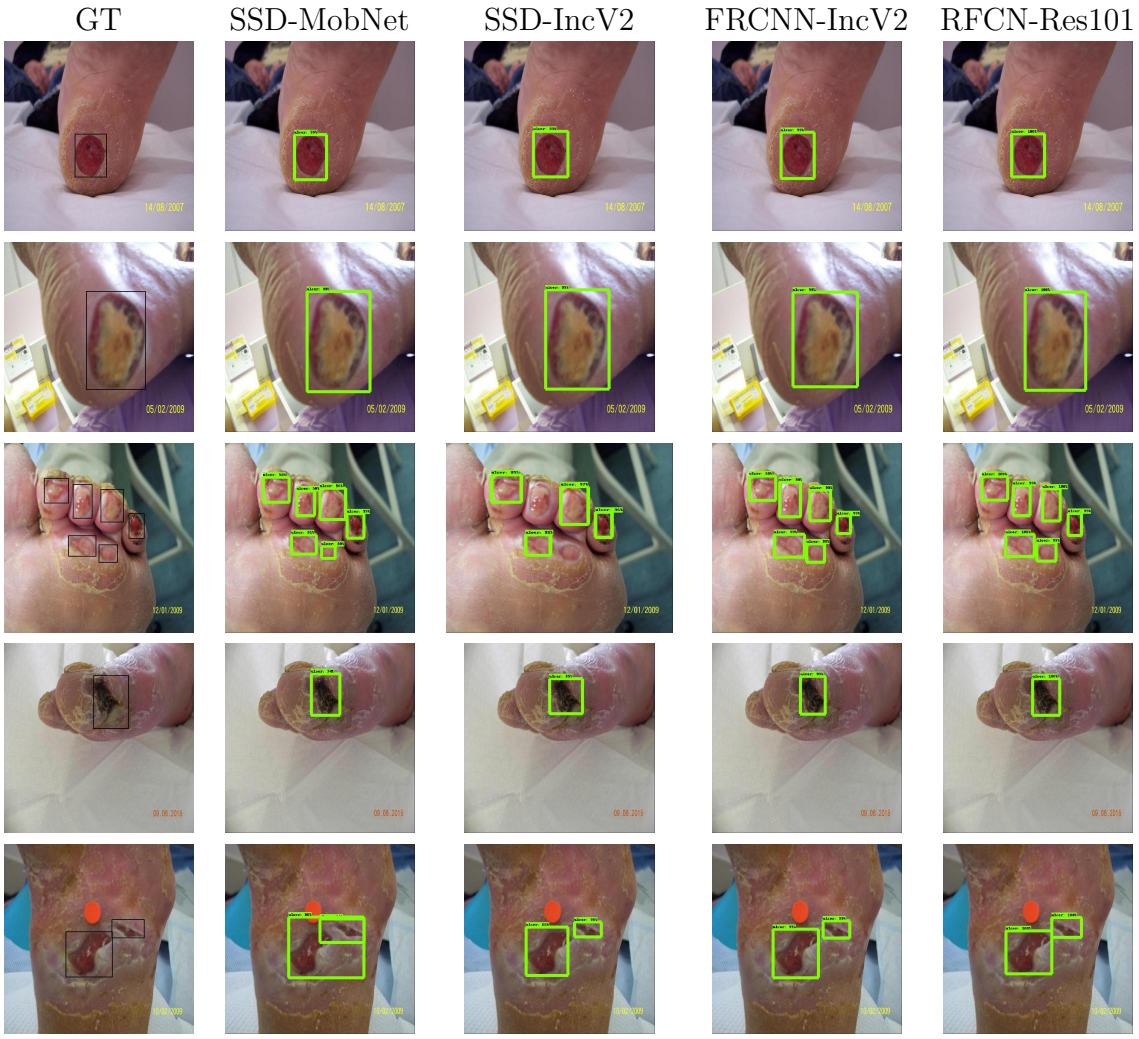


FIGURE 7.7: The accurate localsation results to visually compare the performance of object localisation networks on DFU dataset. Where SSD-MobNet is SSD-MobileNet, SSD-IncV2 is SSD-InceptionV2, FRCNN-IncV2 is Faster R-CNN with InceptionV2, and RFCN-Res101 is R-FCN with ResNet101.

score of 96.1%, which was slightly better than Faster R-CNN with Inception. SSD-InceptionV2 ranked third in both of these performance measure categories with a difference of 4.6% in *Ulcer mAP* and 3.5% in *Overlap Percentage* from the first position. In performance measures, overall Faster R-CNN with InceptionV2 was the best performer, and the most lightweight SSD-MobileNet emerged as the worst performer in terms of accuracy. Finally, we tested models on the dataset of 105 healthy foot images for specificity measure. None of the above-mentioned models produces any DFU localisation on these healthy images.

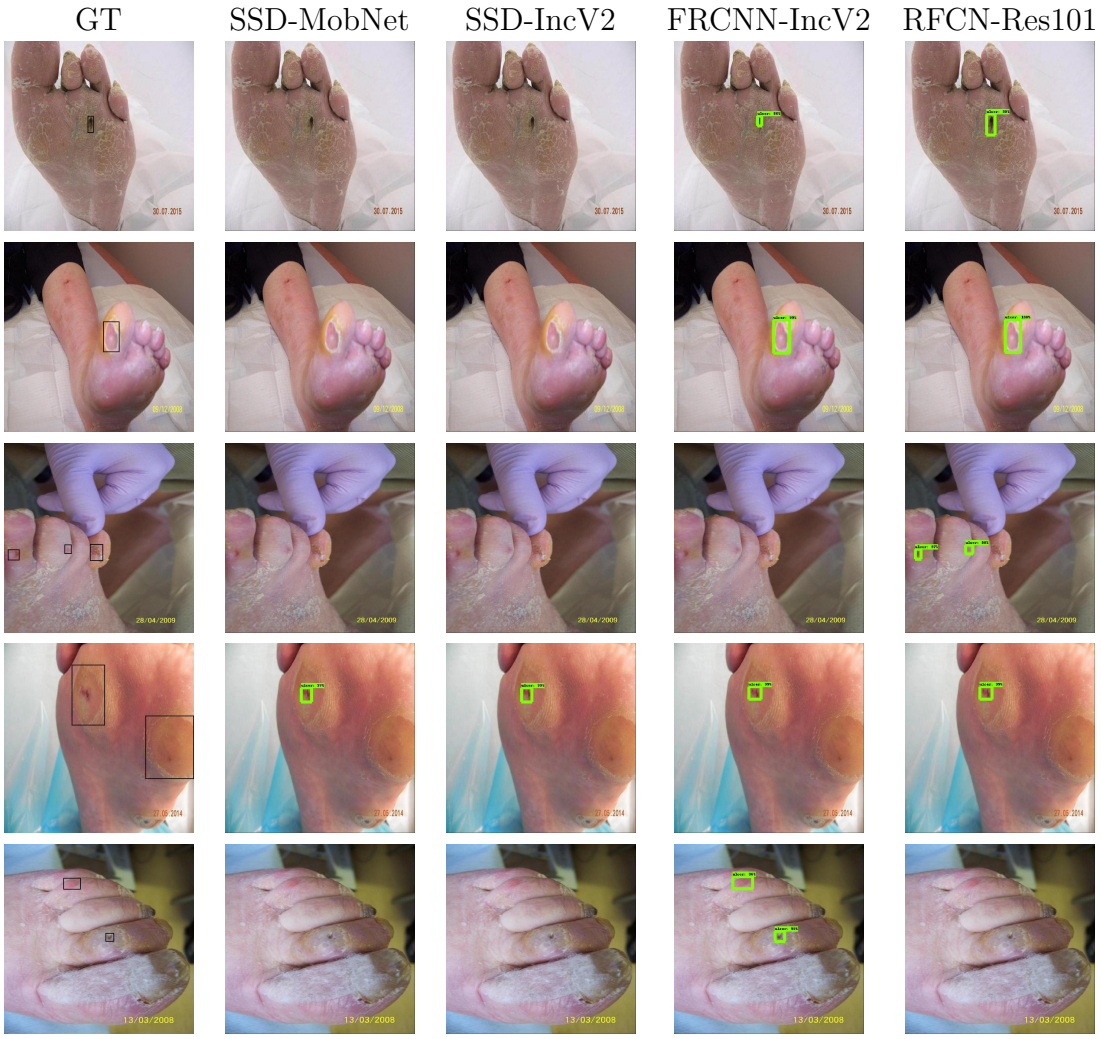


FIGURE 7.8: Incorrect localisation results to visually compare the performance of object localisation networks on DFU dataset. Where SSD-MobNet is SSD-MobileNet, SSD-IncV2 is SSD-InceptionV2, FRCNN-IncV2 is Faster R-CNN with InceptionV2, and RFCN-Res101 is R-FCN with ResNet101.

7.3.1 Inaccurate DFU Localisation Cases

In this work, we explored different object localisation meta-architectures to localise DFU on full foot images. Although the performance of all models is quite accurate as shown in Fig. 7.7, this section explores inaccurate localisation cases by trained models on DFU dataset in 5-fold cross-validation as shown in the Fig. 7.8. We found that trained models were struggled to localise the DFU of very small size and that has a similar skin tone of the foot especially, SSD-MobileNet and SSD-InceptionV2. There are cases of DFU that have very subtle features, not even, most accurate models such as Faster R-CNN with InceptionV2 and R-FCN with ResNet101 were able to detect these conditions.

7.4 Inference of Trained Models on NVIDIA Jetson TX2 Developer Kit

Nvidia Jetson TX2 is the latest mobile computer hardware with an onboard 5-megapixel camera and a GPU card for the remote deep learning applications as shown in the Fig. 7.9. However, it is not capable of training large deep learning models. We installed tensor-flow specifically designed for this hardware to produce inference from the DFU localisation models that we trained on the GPU machine. Jetson TX2 is a very compact and portable device that can be used in various remote locations.

Configuration of Jetson TX2 for Inference (1) Hardware: CPU - dual-core NVIDIA Denver2 + quad-core ARM Cortex-A57, GPU - 256-core Pascal GPU, RAM - 8GB LPDDR4 (2) Software: Ubuntu Linux 16.04 & Tensor-flow.

We did not find any difference in the prediction of the models on Jetson TX2 hardware and the GPU machine; the only let-off is the slow inference speed on the Jetson TX2. It is obviously due to limited hardware compared to the GPU machine. For example, the speed of SSD-MobileNet was 70 ms per inference on Jetson TX2 as compared to 30 ms on GPU machine. Also, for real-time localisation, models can produce the visualisation of maximum 5 fps using the on-board camera with a lightweight model. Fig 7.10 demonstrates the inference using Jetson TX2.

7.5 Real-time DFU localisation with smartphone application

Training and inference of the deep learning frameworks on a smartphone are challenging tasks due to limited resources of a smartphone. Hence, we trained these object localisation frameworks on the desktop with a GPU card. We utilised the whole dataset of 1775 DFU images for further experiments by randomly splitting 90% data in the training set and remaining 10% in the validation set. We trained only Faster R-CNN with InceptionV2 on this dataset because of the best trade-off between the accuracy and the speed. With android studio and tensor-flow deep

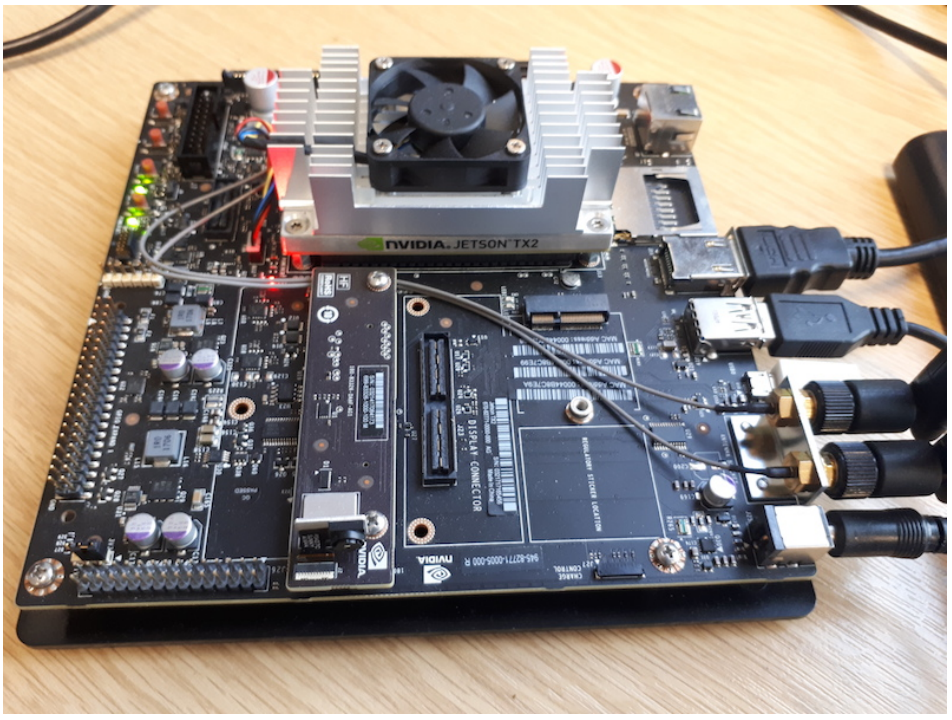


FIGURE 7.9: Nvidia Jetson TX2.

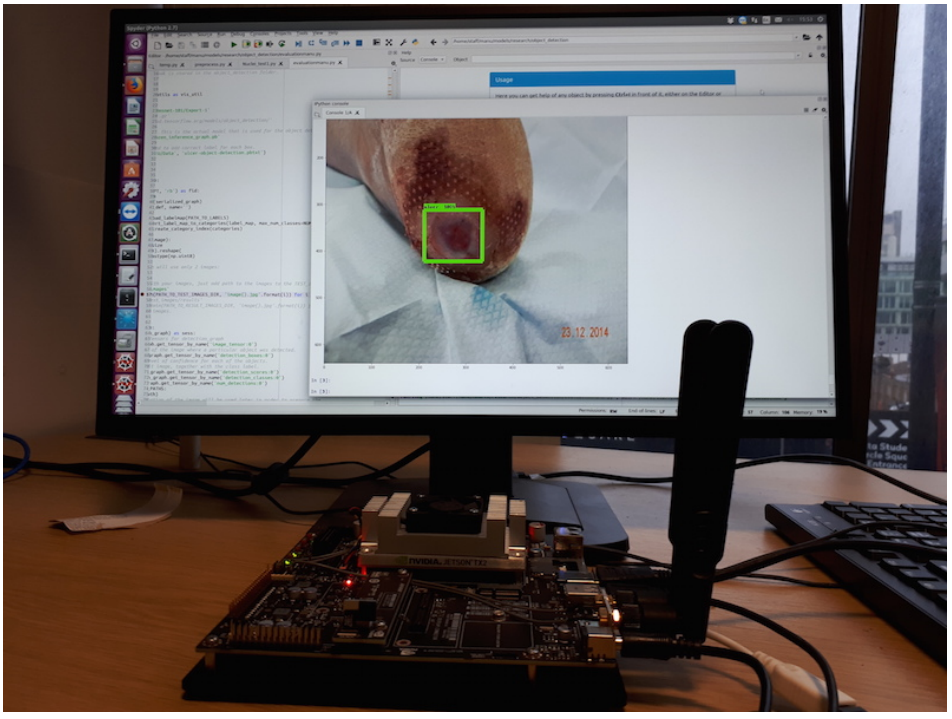


FIGURE 7.10: DFU localisation on Nvidia Jetson TX2 using Faster R-CNN with InceptionV2 on tensor-flow.

FIGURE 7.11: Real-time localisation using smartphone android application. In the first row, images are captured by the default camera. In the second row, the snapshot of real-time localisation by our prototype android application.

learning mobile library, we deployed these models on Samsung A5 2017 (Android Phone) to create the real-time object localisation for DFU. As mentioned in the previous section, we finalised Faster R-CNN with InceptionV2 model for the prototype android application.

We tested our prototype application for the real-time application in real-time healthcare settings as shown in Fig. 7.11. We tested this application on 30 people in this preliminary test in which 10 people were with DFU. Out of 10 people with DFU, our application detected 8 DFU and out of 20 people with normal foot, our application did not detect any false detection. Furthermore, more user-friendly features, care, and guidance will be added to this application to make it a complete package of DFU care for diabetic patients.

7.6 Summary

In this work, we collected an extensive database of 1775 images of DFU. Two medical experts produced the ground truths of this dataset by outlining the region of interest of DFU with an annotator software. Using 5-fold cross-validation, overall, Faster R-CNN with InceptionV2 model using two-tier transfer learning achieved a mean average precision of 91.8%, the speed of 48 ms for inferencing a single image and with a model size of 57.2 MB. To demonstrate the robustness and practicality of our solution to real-time prediction, we evaluated the performance of the models on a NVIDIA Jetson TX2 and a smartphone app. This work demonstrates the capability of deep learning in real-time localization of DFU, which can be further improved with a more extensive dataset.

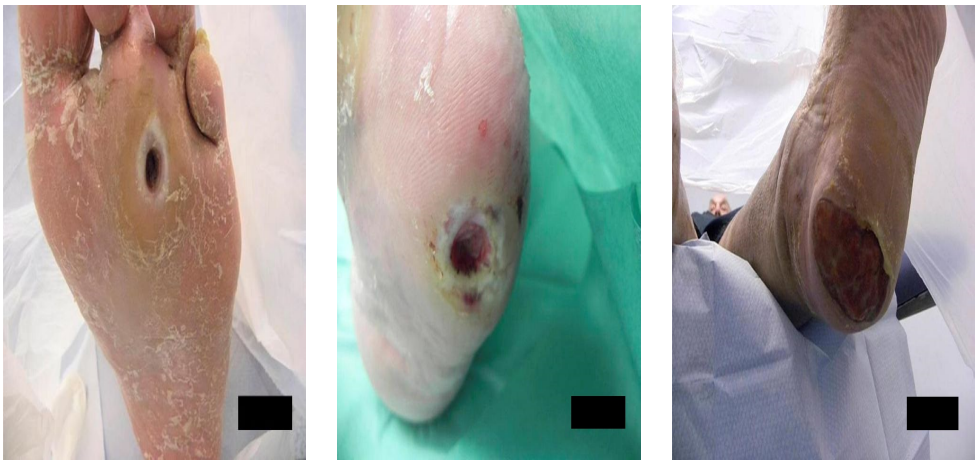
Chapter 8

Detection of Ischemia and Infection in DFU

In this Chapter, we analysed the use of computer vision algorithms to determine the conditions such as area, depth, ischemia, infection in DFU according to the Sinbad classification system on the current dataset. We used various traditional machine learning and deep learning techniques to perform binary classification of ischemia and infection.

8.1 Introduction

The major progress in computer vision allows us to make extensive use of medical imaging data to provide us with better diagnosis, treatment and prediction of diseases [25, 26]. There are numbers of medical classification systems for DFU are discussed such as Wagner, Texas, and Sinbad Classification systems which depend upon the number of factors or conditions that are the site, area, depth, neuropathy, the presence of ischemia, infection [1, 28, 30]. Sinbad classification system is relatively new and simplified classification system introduced by Paul et al. to compare the outcomes of DFU of different populations around the world. Sinbad score stands for S (Site), I (Ischemia), N (Neuropathy), B (Bacterial infection), A (Area), D (Depth). Sinbad scores are relatively easy and better suited for the machine learning algorithms rather than other classification systems as it provides the specific criteria to perform binary classification for each condition of DFU.



Forefoot Midfoot Hindfoot

FIGURE 8.1: Examples of the presence of DFU on. (a) Forefoot, (b) Midfoot and (c) Hindfoot

Applying computer vision techniques to find these conditions or factors for current dataset could be very difficult, as the DFU images are captured in the hospitals without any standardisation that is relative distance and orientation of foot. The current dataset we received with the ethical approval from NHS did not contain any records about these conditions or any medical classification. The predictions of these conditions on DFU images could be very difficult even for experienced podiatrists as there are certain physical and medical tests are needed to assess these conditions. To find the presence or absence of these conditions on DFU, expert annotations from the podiatrists specialised in DFU are required. The brief description of each condition according to the Sinbad scores is described with the computer vision perspective as well.

1. Site: the site of DFU tells about the presence of DFU on which part of the foot. Usually, DFU occurs on the two major sites that are forefoot or midfoot and hindfoot that are shown in Fig. 8.1. Defining the site with computer vision is certainly possible but it can be easily performed by a person even without prior medical knowledge.

2. Area: The area of DFU determines the extent of the 2D shape of DFU on the foot. The area of DFU is classified whether DFU is greater than 1 cm or not as shown in Fig. 8.2. Since, as mentioned earlier, the inconsistent images in the current dataset due to distance, orientation and lighting as the data captured in hospital, DFU images are captured with different magnification and angles as shown in the Fig. 8.3.

Area greater than 1cm

Area less than or equal to 1cm

FIGURE 8.2: Examples of classification of area of DFU

FIGURE 8.3: Example of DFU images are captured with different magnification and angles

3. Depth: the depth of DFU determines the distance from the surface of the foot to the bottom due to tissue damage and loss. The depth of DFU can be classified into two categories whether DFU is superficial that is confined to the skin and subcutaneous tissue or DFU reaching muscle, tendon or deeper as shown in Fig. 8.4. 2D Photo documentation provided in the current dataset cannot accurately measure the depth of DFU.

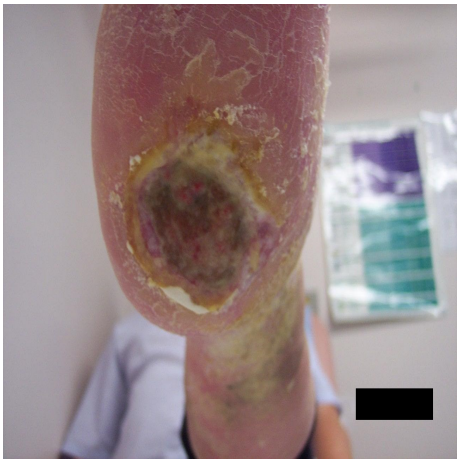
4. Ischemia: DFU appear due to the damage that raised blood sugars can cause sensation and blood circulation. The inadequate blood supply to the foot can lead to a condition called ischemia. The visual appearance of ischemia could be determined with the presence of a pale looking ulcer, or black gangrenous toes (tissues death to part of the foot) as shown in the Fig. 8.5. In computer vision perspective, it is an important hint of the presence of ischemia in the DFU.

Depth reaching tendon

Superficial depth

FIGURE 8.4: Examples of classification of depth of DFU

Presence of ischemia in DFU



No ischemia in DFU

FIGURE 8.5: Cases of the presence of ischemia and no ischemia in DFU in foot images

5. Infection: Infection is defined as bacterial soft tissue or bone infection in the DFU which is based on the presence of at least two classic findings of inflammation or purulence as shown in Fig. 8.6. It is very hard to determine the presence or absence of diabetic foot infections in DFU images because, in the medical system, blood testing is performed as supporting evidence. Also, in this dataset, the images are captured after the debridement of necrotic and devitalised tissues which might be an important indicator of the presence of infection in DFU.

6. Neuropathy: Neuropathy is defined as loss of sensation in the lower extremities i.e. foot region due to damage of the peripheral nerves. Neuropathy is again infeasible with the help of computer vision techniques as there is no

Presence of infection No Infection

FIGURE 8.6: Cases of presence of infection and no ischemia in DFU in foot images

visual hint to detect neuropathy in the foot. But, the patients can determine the neuropathy condition with the help of very simple physical procedures. Usually, patients with DFU have certain neuropathy condition.

This work focuses on finding the presence or absence of ischemia and infection in DFU of foot images as detecting other conditions are not feasible with computer vision techniques due to different factors such as non-standardised dataset, 2D images and requirement of physical and medical tests to determine certain conditions as mentioned above.

In the related work, Netten et al. [46] find that clinicians achieved low validity and reliability for remote assessment of DFU in foot images. Hence, it is clear that analysing these conditions on the images are extremely difficult even by the expert podiatrists. In various image recognition and natural language processing tasks where machine learning algorithms can perform better than skilled humans. This experiment is performed to analyse the performance of machine learning algorithms on the detection of ischemia and infection in DFU images.

8.2 Methodology

This section describes proposed methods for Natural Data-Augmentation, feature descriptors and classifiers used for the traditional machine learning. Brief description of deep learning methods and experimental settings is also discussed in this

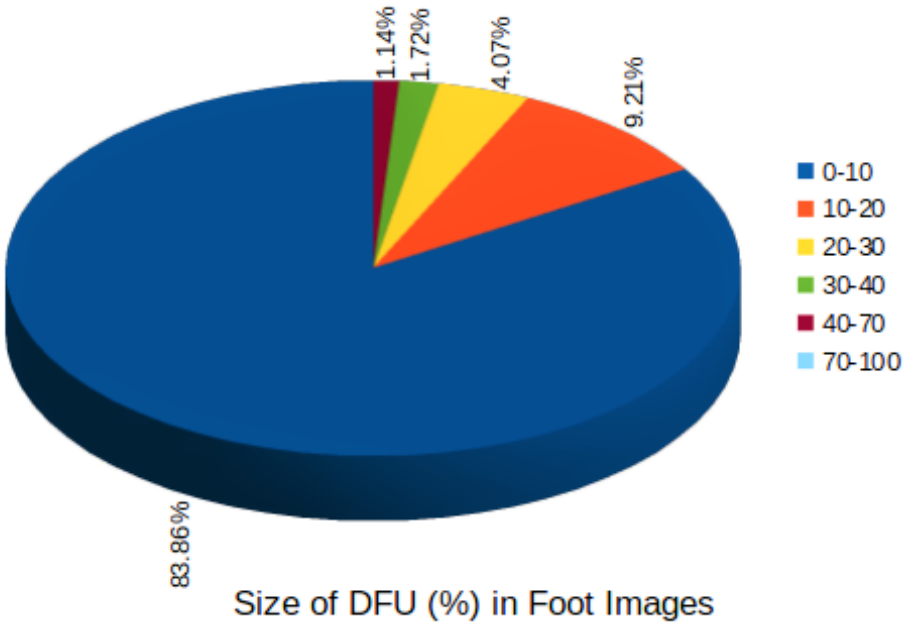


FIGURE 8.7: Comparison of Size of DFU against the size of image in the DFU dataset of 1459 images

section.

8.2.1 Natural Data-Augmentation for DFU images

In the DFU dataset, the size of images varies between 1600 × 1200 and 3648 × 2736 depending on the different professional cameras used to capture the data. In deep learning, data augmentation is tipped as an important tool to improve the performance of algorithms.

As shown in Fig. 8.7, about 92% of DFU have area between 0% to 20% on foot images. In common data-augmentation, there is the number of techniques used such as flip, rotation, random scale, random crop, translation, Gaussian noise to perform augment in the dataset. Since DFU occupy very small percentage of the total area of foot images, there is a risk of missing the region of interests by using important augmentation technique such as random scale, crop and translation. Hence natural data-augmentation is more suitable for the DFU evaluation rather than common data-augmentation.

To focus more on ROI of DFU, we proposed the use of automatic data augmentation technique called natural data-augmentation which is based on DFU localization using Faster R-CNN [27, 58]. This augmentation technique helps in

TABLE 8.1: Performance measures of object localisation models on DFU dataset

Model Name	*Speed (ms)*	*Size of Model (MB)*	*Ulcer mAP*	*Overlap Percentage (%)*
SSD-MobileNet	28	22.6	84.9	89.4
SSD-InceptionV2	37	53.5	87.2	92.6
Faster R-CNN with InceptionV2	48	52.2	91.8	95.8
R-FCN with Resnet 101	90	199.1	90.6	96.1
Faster R-CNN with Inception ResNet V2	626	596.7	92.9	96.3

assisting the machine algorithms to clearly pinpoint ROI of foot images and focus on finding the strong features exists in this area.

8.2.2 Proposed method for Natural Data-Augmentation

First of all, we used the deep learning based localisation method called Faster R-CNN with InceptionResNetV2 to get the ROI of DFU on foot images in our dataset as shown in Fig. 8.8. This method further improved the performance of localisation methods from the previous chapter as shown in Table 8.1. Our proposed method can provide a robust natural data-augmentation technique for DFU images as shown in Fig. 8.9 by removing the unnecessary background data and without any particular loss of quality (only in the case of very small ulcers). As most of the deep learning algorithms use smaller image size as input from 224×224 to 331×331 depending on the architecture. The number of natural data-augmentation with localisation methods depends upon the input image size for algorithm and the ratio between the size of ROI and size of an image. In Fig. 8.9 and 8.10, we showed natural data-augmentation with different magnification and angles using our proposed methods.

8.2.3 Traditional Machine Learning

We investigated the use of human design features with TML on the binary classification of infection and ischemia. We used the color descriptors as mentioned before that could be the important visual cues for identification of ischemia and infection in DFU. First of all, we used SLIC superpixels technique to produce superpixel oversegmentation of DFU patches [137] and then computed mean RGB color of each superpixel as shown in Fig. 8.11. Finally, with different threshold values from

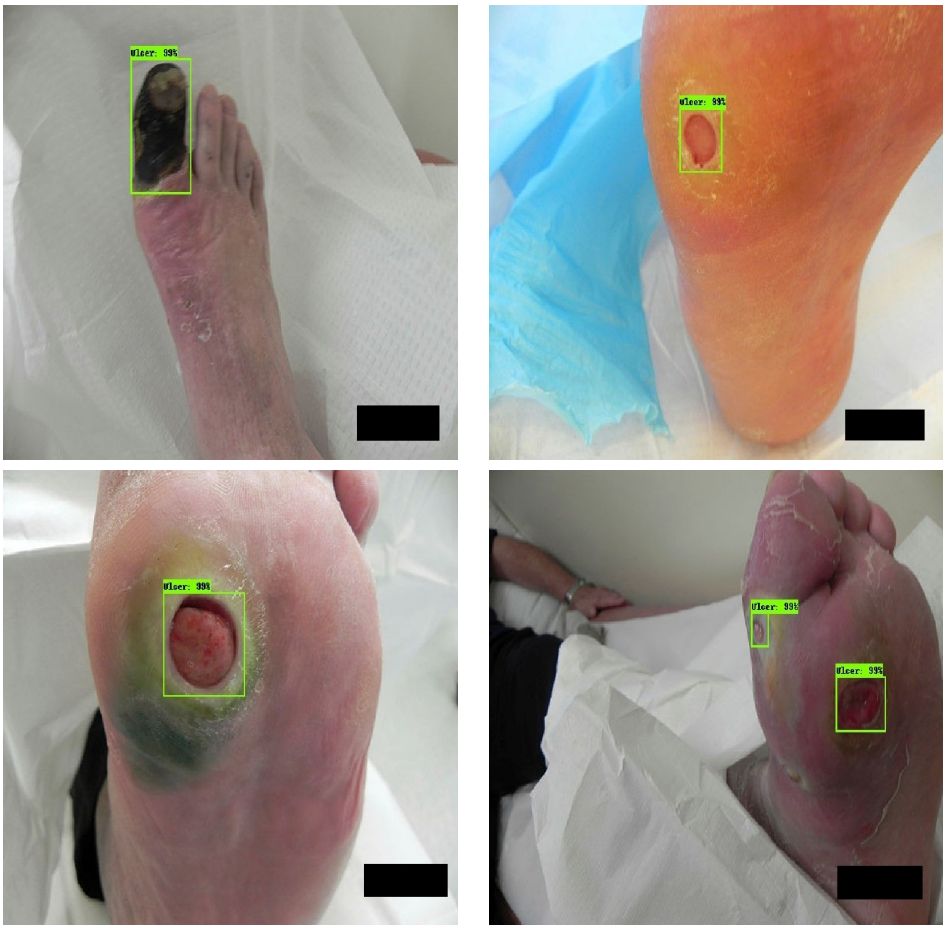
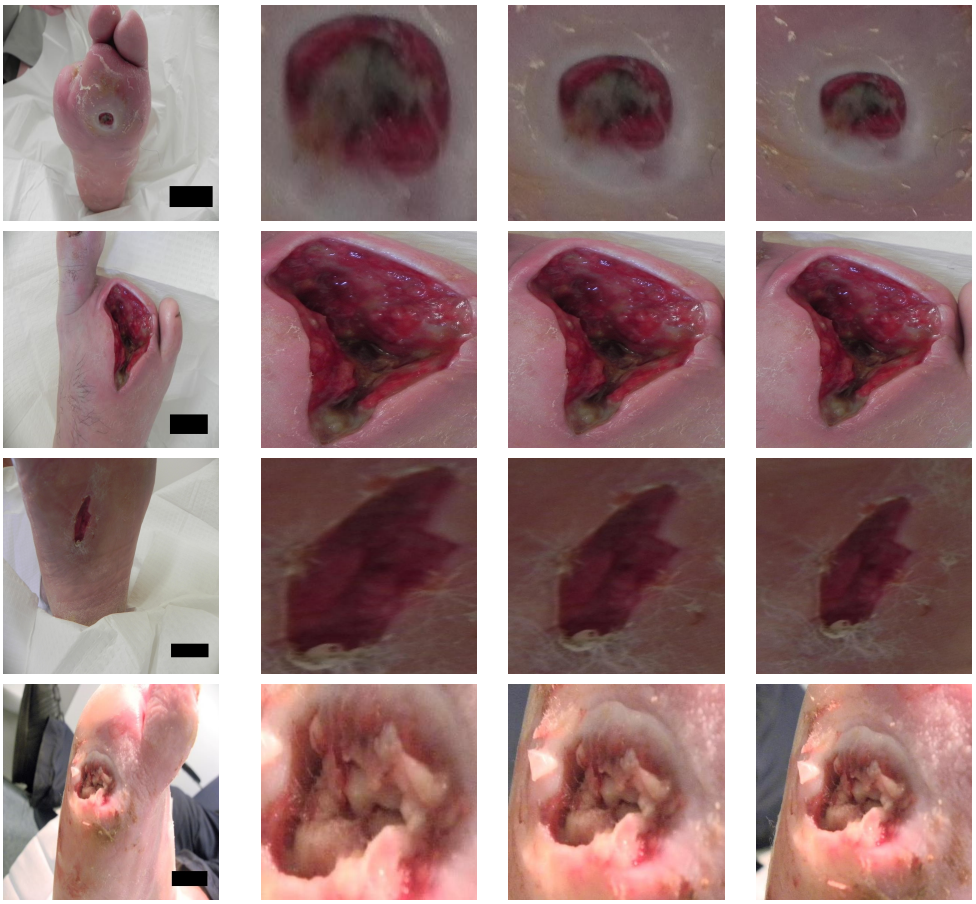


FIGURE 8.8: Examples of DFU detection and localisation using Faster-RCNN with Inception ResNet V2

each color channel, we extracted regions of two particular colors of interest that are red and black from the DFU patches as shown in Fig. 8.12. For these classification problems, we tried number of classifiers with standard hyper-parameters on these color features in which BayesNet, Random Forest, and Multilayer Perceptron were selected as these methods achieved the highest accuracy among other machine learning classifiers [138–143].

8.2.4 Convolutional Neural Networks

For comparison with the traditional features, deep learning algorithms are used to perform binary classification to classify (1) infection and non-infection; (2) ischemia and non-ischemia classes in DFU patches. For this work, we fine-tuned (transfer learning from pre-trained models) the state-of-the-art CNN models such as Inception-V3, ResNet50, and InceptionResNetV2 for this task.



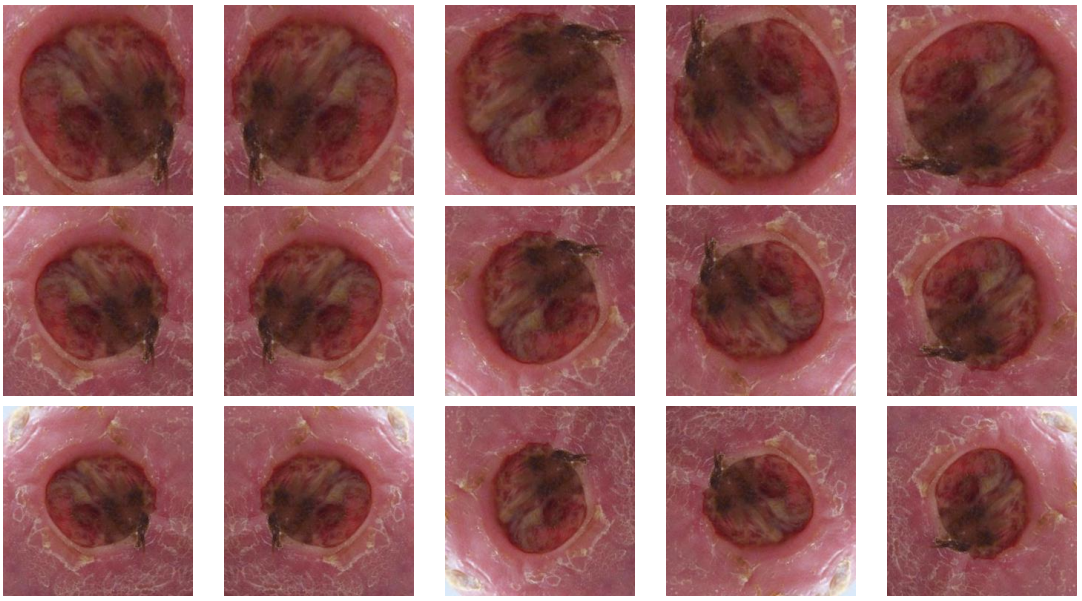
(a) Original Image (b) Ist MAG (c) 2nd MAG (d) 3rd MAG

FIGURE 8.9: Natural data-augmentation produced from the original image with different magnifications. MAG refers to magnification

Inception-V3 is a new iteration of the original inception architecture designed by Google team with new features such as factorisation of bigger convolution kernels to multiple smaller convolution kernels and improved normalisation. In this network, depth-wise separable convolutions are used in initial layers of architecture to reduce the computations of down-sampling the input images. They also introduced batch normalisation layer which can decrease internal covariate shift, also combat the gradient vanishing problem to improve the convergence during training [133, 144].

ResNet50 is a lighter residual learning network version of ResNet101 which won the first place on ILSVRC 2015 classification task [120]. The core idea of ResNet is providing a shortcut connection between layers to gain maximal representation from both initial as well as later layers in training of the network.

InceptionResNetV2 is a very deep network which combines the strengths of



(a) Image (b) Mirror (c) 90° (d) 180° (e) 270°

FIGURE 8.10: Natural data-augmentation of different angles produced from the images (different magnification)

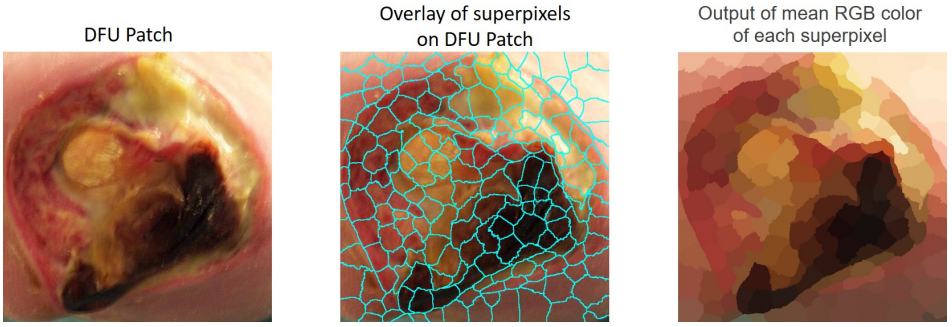


FIGURE 8.11: Example of superpixel oversegmentation and computing the mean RGB color of each superpixel in DFU patch.

both inception and residual learning networks as the name suggests. It is inspired by InceptionV3 architecture with residual connections between the layers to successfully train even deeper neural networks, which have to lead to even better performance. It achieved new state-of-the-art results in terms of accuracy on various standard datasets [133, 134].

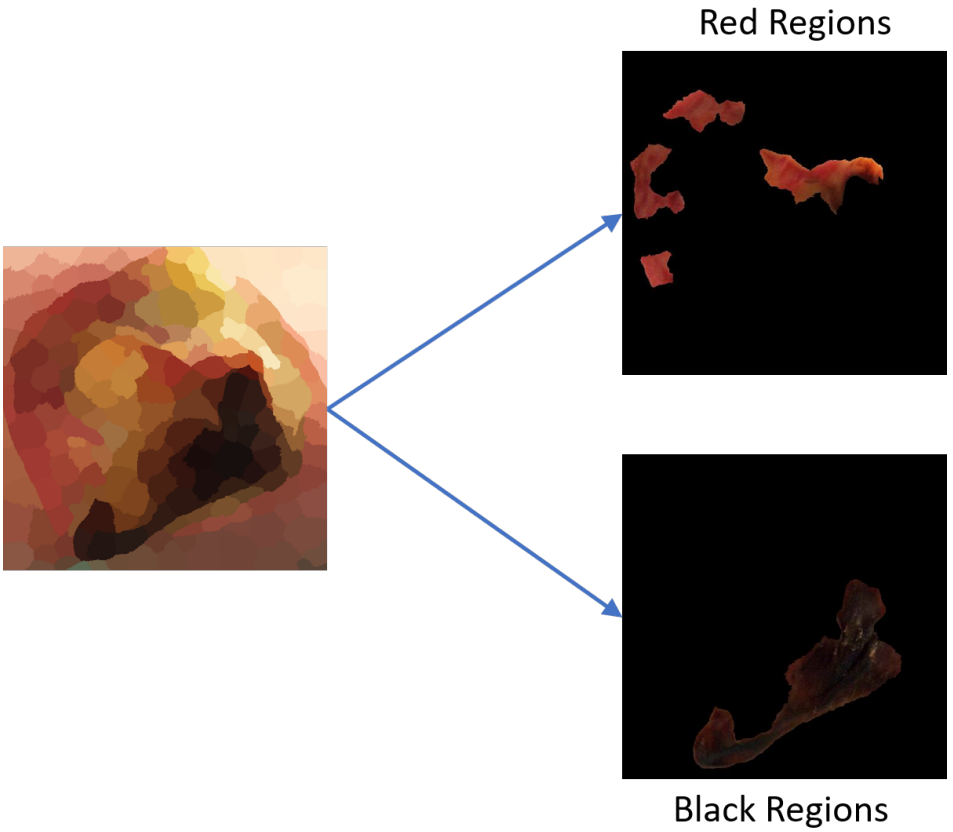


FIGURE 8.12: Example of extracting red and black regions from DFU patch with different threshold values

8.3 Results and Discussion

Both infection and ischemia datasets were split into the 70% training, 10% validation and 20% testing sets and we adopted the 5-fold cross-validation technique. Hence, in ischemia dataset, for training, validation, and testing set using the proposed methods, we used approximately 6909 patches, 987 patches, and 1974 patches in training, validation, and testing sets respectively whereas in infection dataset, we used 4124 patches, 589 patches, and 1179 patches from the 1459 original foot images. As mentioned previously, we used both TML models and CNNs models to do the classification task and utilised 256×256 RGB images as input for CML and InceptionV3, AlexNet, and ResNet50. For InceptionResNetV2, we resized the dataset to 299×299.

In Table 8.2 and 8.3, we report *Accuracy*, *Sensitivity*, *Precision*, *Specificity*, *F-Measure* and *MCC* as our evaluation metrics. In medical imaging, *Sensitivity* and *Specificity* are considered reliable evaluation metrics for classifier completeness.

TABLE 8.2: The performance measures of binary classification of Ischemia by both traditional machine learning and CNNs where MCC is Matthew Correlation Coefficient

	Accuracy	*Sensitivity*	*Precision*	*Specificity*	*F-Measure*	*MCC Score*
BayesNet	0.785±0.022	0.774±0.034	0.809±0.034	0.800±0.027	0.790±0.020	0.572±0.044
Random Forest	0.780±0.041	0.739±0.049	0.872±0.029	0.842±0.034	0.799±0.033	0.571±0.078
Multilayer Perceptron	0.804±0.022	**0.817±0.040**	0.787±0.046	0.795±0.031	0.800±0.023	0.610±0.045
InceptionV3 (CNN)	0.841±0.017	0.784±0.045	0.886±0.018	0.898±0.022	0.831±0.021	0.688±0.031
ResNet50 (CNN)	**0.862±0.018**	0.797±0.043	**0.917±0.015**	**0.927±0.017**	**0.852±0.022**	**0.732±0.032**
InceptionResNetV2 (CNN)	0.853±0.021	0.789±0.054	0.906±0.017	0.917±0.019	0.842±0.027	0.714±0.039

TABLE 8.3: The performance measures of binary classification of Infection task by both traditional machine learning and CNNs results. where MCC is Matthew Correlation Coefficient

	Accuracy	*Sensitivity*	*Precision*	*Specificity*	*F-Measure*	*MCC Score*
BayesNet	0.639±0.036	0.619±0.018	0.653±0.039	0.660±0.015	0.622±0.079	0.290±0.070
Random Forest	0.605±0.025	0.608±0.025	0.607±0.037	0.601±0.069	0.606±0.012	0.211±0.051
Multilayer Perceptron	0.621±0.026	0.680±0.023	0.622±0.057	0.570±0.023	0.627±0.074	0.281±0.055
InceptionV3 (CNN)	0.662±0.014	**0.693±0.038**	0.653±0.015	0.631±0.034	0.672±0.019	0.325±0.029
ResNet50 (CNN)	0.673±0.013	0.692±0.051	0.668±0.023	0.654±0.051	0.679±0.019	0.348±0.028
InceptionResNetV2 (CNN)	**0.676±0.015**	0.688±0.052	**0.672±0.015**	**0.664±0.039**	**0.680±0.024**	**0.352±0.031**

When comparing the performances, the methods including TML and CNN performed better in the binary classification of ischemia than infection. The average performance of all the models in terms of accuracy in ischemia dataset is 82.1% which is significantly higher than average accuracy of 64.6% in infection dataset. MCC score is considered to be a viable performance measure for the different machine learning approaches for classification, with an average *MCC Score* for ischemia classification of 64.8% is higher compared to the infection classification of 30.1%. When comparing the performances of TML and CNNs, CNNs (85.2%) outperformed the TML models (79%). Similarly, in infection classification, the accuracy of CNNs (67%) performed better than TML (62.1%) with a margin of 4.9%.

In ischemia classification, ResNet50 received highest score in all performance

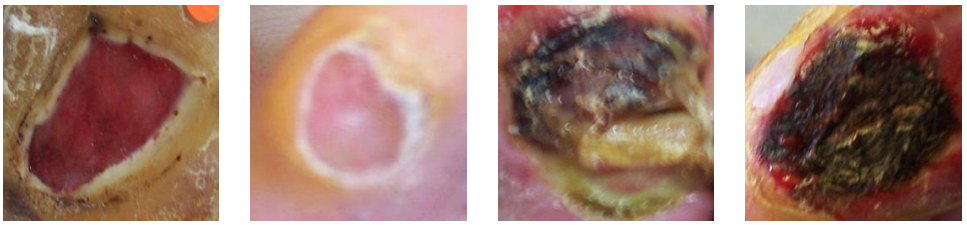
measures except for *Sensitivity* in which TML method multilayer perceptron received a score of 81.7% but scored lowest score of 79.5% in *Specificity*. For *Specificity*, the CNN methods performed extremely well with average score of 91.4% when compared to TML methods with average score of 81.2%. There is a huge margin of 13.2% between the highest result (ResNet50) and the lowest result (Multilayer perceptron). There is a more significant gap of approximately 16.1% in *MCC Score* for the methods performance, with results ranging from 57.1% to 73.2%.

In infection classification, both TML and CNN methods received moderate score in the performance measures. Similarly, CNN methods once again performed better than TML methods achieving highest score in all performance measures. The InceptionResnNetV2 marginally performed better than other CNN classifiers especially in *Specifcity* with score of 66.4% in infection classification. For *Sensitivity*, all the CNNs performed equally well with InceptionV3 achieved the highest score of 69.3%. For TML methods, Multilayer Perceptron performed well in *Sensitivity*, whereas BayesNet in *Specificity* and *Precision*.

ResNet50 is the best performers for various evaluation metrics among all the classifiers in ischemia classification whereas InceptionResNetV2 performance is best in infection classification.

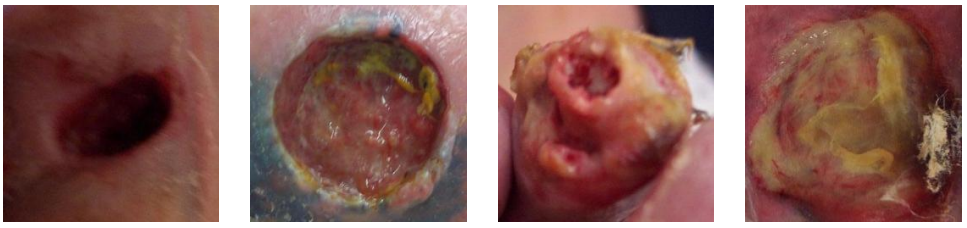
8.3.1 Experimental Analysis and Discussion

Analysis of conditions of DFU with the computerised methods is very important for the limited medical experts and healthcare settings. This preliminary experiment of binary classification of ischemia and infection of DFU is performed in this work. The main motivation of this experiment to find what conditions of ischemia and infection are at high risk of being misclassified by computer vision algorithms. Few examples of correctly and incorrectly classified cases in both binary classifications of ischemia and infection are illustrated in Fig. 8.13, 8.14, 8.15, and 8.16. As for the misclassified cases, there are huge intra-class dissimilarities and inter-class similarities between (1) infection and non-infection; (2) ischemia and non-ischemia cases in the DFU that make classifiers difficult to predict the right class. Also, there are other influential factors in the classification of these conditions such as lighting conditions, marks, tattoo and skin tone due to the patient's ethnicity. In misclassified cases of non-ischemia as shown in Fig. 8.14, the cases (a) and (b) are



(a) (b) (c) (d)
Accurate non-ischemia cases Accurate ischemia cases

FIGURE 8.13: Correctly classified patches by InceptionResNetV2 on Ischemia dataset. (a) and (b) represents non-ischemia cases. (c) and (d) represents ischemia cases.

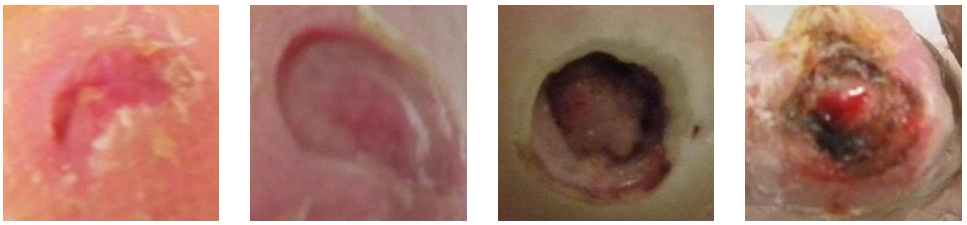


(a) (b) (c) (d)
Misclassified non-ischemia cases Misclassified ischemia cases

FIGURE 8.14: Misclassified patches by InceptionResNetV2 on Ischemia dataset. (a) and (b) represents non-ischemia cases. (c) and (d) represents ischemia cases.

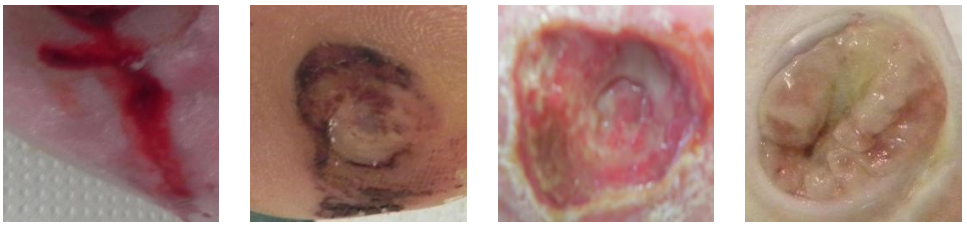
hindered by the lighting condition and tattoo respectively whereas in the (c) and (d) misclassified ischemia cases, the ischemia features are too subtle to be detected by the algorithm. In Fig. 8.16, misclassified cases of non-infection, the presence of blood in the case (a) whereas in the case (b) belongs to one of the rare cases in the dataset that is the presence of ischemia and non-infection. In misclassified infection cases, the visual indicators of infection in these cases were too subtle.

The current ground truths are based on visual inspection by experts only and not supported by the medical notes or clinical tests. Also, DFU images were derided with debridement before these images were captured. Hence, the debridement of DFU removed the important visual indicators of infection such as coloured exudate. Therefore, the sensitivity and specificity of these algorithms can be further improved in the future feeding in ground truth from clinical tests such as vascular assessments (ischemia) and blood tests (to identify the presence of any bacterial infection).



(a) (b) (c) (d)
Accurate non-infection cases Accurate infection cases

FIGURE 8.15: Correctly classified patches by InceptionResNetV2 on Infection dataset. (a) and (b) represents non-infection cases. (c) and (d) represents infection cases.



(a) (b) (c) (d)
Misclassified non-infection Misclassified infection cases

FIGURE 8.16: Misclassified patches by InceptionResNetV2 on Infection dataset. (a) and (b) represents non-infection cases. (c) and (d) represents infection cases.

8.4 Summary

In this work, we trained various classifiers based on traditional machine learning algorithms and CNNs to discriminate the conditions of (1) ischemia and non-ischemia; (2) infection and non-infection in DFU skin. We found high-performance measures in the binary classification of ischemia, whereas moderate performance by classifiers in the classification of infection and non-infection classes. It is vital to understand the features of both conditions of DFU (ischemia and infection) in the computer vision perspective. Determining these conditions especially infection from the non-standard foot images could be very challenging due to (1) high visual intra-class dissimilarities and inter-class similarities between classes; (2) the visual indicators of infection and ischemia are too subtle in DFU; (3) medical tests are needed to assess these conditions; (4) other factors such as lighting conditions, marks, and skin tone due to patient's ethnicity. Ground truths enhanced by clinical tests for the ischemia and infection may provide further insight and further improvement of algorithms even where there is no apparent visual indicator by eye. In the case of infection even after debridement, ground truth informed by blood tests for infection may yield improvements to sensitivity and specificity

even in the absence of overtly obvious visual indicators. With more balanced data and improved data capturing of DFU, the performance of these methods could be improved in the future. This work has the potential for technology that may transform the detection and treatment of diabetic foot ulcers and lead to a paradigm shift in the clinical care of the diabetic foot.

Chapter 9

Conclusion and Future Works

In this final Chapter, a summary of the contributions of this thesis on recognition and analysis of DFU are discussed. A critical analysis of the work completed is done with a focus on the strengths and limitations found during the research. It also highlights potential future improvements to this field and the direction in which it is heading for researchers in this continuously growing area.

9.1 Research Findings

A summary of the research objectives is shown in Table 9.1 along with the corresponding outcomes. These findings will detail the reason for each objective and how the outcome was achieved.

Diagnosis and recognition of DFU by the computerised method has been an emerging research area with the evolution of computer vision, especially deep learning methods. In this work, we investigated the use of both conventional machine learning and deep learning for the recognition and analysis DFU. We achieved relatively good performance using a conventional machine learning technique. But, due to multiple intermediate steps, this approach is very slow for DFU recognition tasks. In deep learning, we used different architectures to train the end-to-end models on the DFU dataset with different hyper-parameter settings to detect DFU on the full foot images with high accuracy. These methods are capable of localising and segmenting multiple DFU with high inference speed. Then, we

TABLE 9.1: The research objectives (defined in Section 1.4) against the actual outcomes.

No.	Objective	Outcome
1	To study the literature related to the background of DFU, medical classification systems for DFU, and computerised methods for recognition of DFU of various grades and stages.	We identified the research gaps in computerized methods for recognition of DFU, discussed various popular medical classification systems used to grade DFU and established standardised DFU datasets (with experts annotation) for popular computer vision tasks that are classification, segmentation and localisation.
2	To propose a novel computer vision method for DFU classification based on deep learning approach to differentiate normal skin lesions and DFU skin lesion in the foot region.	DFU dataset of 292 images is delineated by experts to produce healthy skin and DFU skin patches. We used machine learning algorithms to extract the features for DFU and healthy skin patches to understand the differences in the computer vision perspective. A novel deep learning classification framework is introduced - DFUNet, which outperformed the state-of-the-art traditional machine learning and deep learning methods for DFU classification [2].
3	To develop new CNN-based automatic segmentation methods to segment DFU and surrounding skin on full foot images as surrounding skin is an important visual indicator to assess the progress of DFU.	Experts precisely delineated the DFU and the surrounding skin region in full foot images. This is the first time, segmentation of surrounding skin is performed which is an important indicator for clinicians to assess the progress of DFU. We proposed to use two-tier transfer learning segmentation methods for semantic segmentation of DFU and its surrounding skin [3].

TABLE 9.2: The research objectives (defined in Section 1.4) against the actual outcomes.

No.	Objective	Outcome
4	To develop robust and lightweight deep learning methods for DFU localisation that can be utilized in mobile devices for remote monitoring.	State-of-the-art deep learning localisation methods are tested on the extensive DFU dataset of 1775 images and FootSnap dataset. We transferred the robust and lightweight models on mobile devices such as Nvidia Jetson TX2 and smart-phone android application for remote monitoring of DFU [1].
5	To analyse the different conditions of diabetic foot pathologies according to the popular medical classification systems.	We investigated the different conditions of DFU such as site, infection, neuropathy, bacterial infection, area, and depth according to the computer vision perspective. In this work, we used machine learning algorithms to determine the important conditions of DFU such as bacterial infection and ischemia.

demonstrated how the localisation methods can be easily transferred to a portable device, Nvidia Jetson TX2, to produce inference remotely. Finally, these deep learning methods were used in android application to provide real-time DFU localisation. In this work, we developed mobile systems that can assist both medical experts and patients for the DFU diagnosis and follow-up in the remote settings. In the later experiment, we used the proposed natural data-augmentation with the help of DFU localisation to create DFU patches from full size foot images. These patches are useful to focus more on finding the important characteristics of DFU such as infection and ischemia. Then, we investigated the use of both CML and CNNs to classify these conditions as binary classification. In this experiment, we received very good performance when it comes to find ischemia despite the unbalanced dataset. But in the case of infection, the classifiers did not perform well, as the condition of infection is very hard to recognise from the foot images even by the experienced podiatrists [46].

Despite receiving very good accuracy with different algorithms proposed in

terms of classification, segmentation and localization methods, there were few limitations regarding recognition of DFU in some particular cases such as pre-ulcer conditions and very small DFU with subtle features. The current DFU dataset was captured from Lancashire Teaching Hospital, where most of the DFU images are captured with already significant developed of DFU. There were very few cases in which pre-ulcer and subtle DFU were captured. Hence, there is a need for more cases of DFU of these grades in the DFU dataset in order to make algorithms more robust to detect these particular DFU.

Developing the remote, computerised and innovative DFU diagnosis system according to the medical classification systems and exactness accomplished by the podiatrist, it demands a significant amount of research. To assist podiatrist, foot analysis with computerised methods in the near future, the following issues need to be addressed.

1. The recognition of DFU on foot images with computerised methods is a difficult task due to high inter-class similarities and intra-class variations in terms of color, size, shape, texture and site amongst different classes of DFU. Although, recognition of DFU on full foot images is a valuable study, further analysis of each DFU on foot images is required according to the medical classification systems followed by podiatrists such as Texas Classification of DFU [1] and SINBAD Classification System [30]. We presented the analysis of computer vision techniques to determine important conditions such as infection and ischemia. The current dataset is not suitable for finding other conditions such as area, and depth.

2. Ground truths enhanced by clinical tests for the ischemia and infection may provide further insight and further improvement of algorithms even where there is no apparent visual indicator by eye. In the case of infection even after debridement, ground truth informed by blood tests for infection may yield improvements to sensitivity and specificity even in the absence of overtly obvious visual indicators. The vascular assessments may be useful for recognition of ischemia.

3. Most of the state-of-the-art computerised imaging methods rely on supervised learning. Hence, there is a need for laborious manual annotation by medical experts according to these popular classification systems. For example, Texas classification system classifies DFU into 16 classes depending

on conditions of DFU based on ischemia, infection, area and depth. These methods can be extended to produce localisation of DFU and determine the outcome of DFU according to the Texas classification system with substantial image data belonging to each class and expert annotations.

4. Deep learning methods require a considerable amount of data to learn features of abnormality in medical imaging. To achieve accurate DFU recognition according to different classification systems, multiple images of same DFU covering key specific conditions such as lighting conditions, the distance of image capture from the foot and orientation of the camera relative to the foot. To our best knowledge, there are no publicly available standardised DFU dataset with descriptions and annotation. Hence, there is a requirement of publicly available annotated DFU dataset with essential diagnostic in this regard. The standardised dataset can help to produce even more accurate results with these methods.

5. Early detection of key pathological changes in the diabetic foot leading to the development of a DFU is really important. Hence, the time-line dataset of patients with early signs of DFU until the diagnosis is required to achieve this objective. With these methods and time-line dataset, the early prediction, healing progress and other potential outcomes of DFU could be possible.

6. The DFU diagnosis system should be scalable to multiple devices, platforms and operating systems.

In the present situation, manual inspection by podiatrists remains the ideal solution for the diagnosis of DFU as computer vision and the current dataset is ineffective in determining the conditions of DFU such as depth, area, neuropathy. Also, Netten et al. [46] claimed that human observers achieved low validity and reliability for remote assessment of DFU. Hence, with the help of improved dataset and better algorithms, the computerised diagnosis system could be used as a tool to improve human performance.

9.2 Future Works

The current DFU dataset, the images are captured with different orientation and distance as shown in Fig. 9.1. Hence it is very hard to estimate the approximate

FIGURE 9.1: DFU images of same foot are captured with different magnification and angles

FIGURE 9.2: Future work consists of finding an approximate size and site of DFU

area of DFU. Site of DFU is also considered as one of the important condition in SINBAD classification system to predict the outcome of DFU. Our future work would emphasize finding the site and approximate area of DFU irrespective of orientation and distance as shown in the Fig 9.2.

With limited human resources and facilities in healthcare systems, DFU diagnosis is a significant workload and burden for the healthcare systems. The computer-based systems have huge potential to assist healthcare systems in the DFU assessment. The primary focus of this thesis to develop automatic computer vision methods for robust recognition of DFU. In the last chapter, we further analyse the important conditions that are infection and ischemia with machine learning algorithms. Another future target is to build a complete computerized DFU diagnosis system that can determine the important conditions such as site,

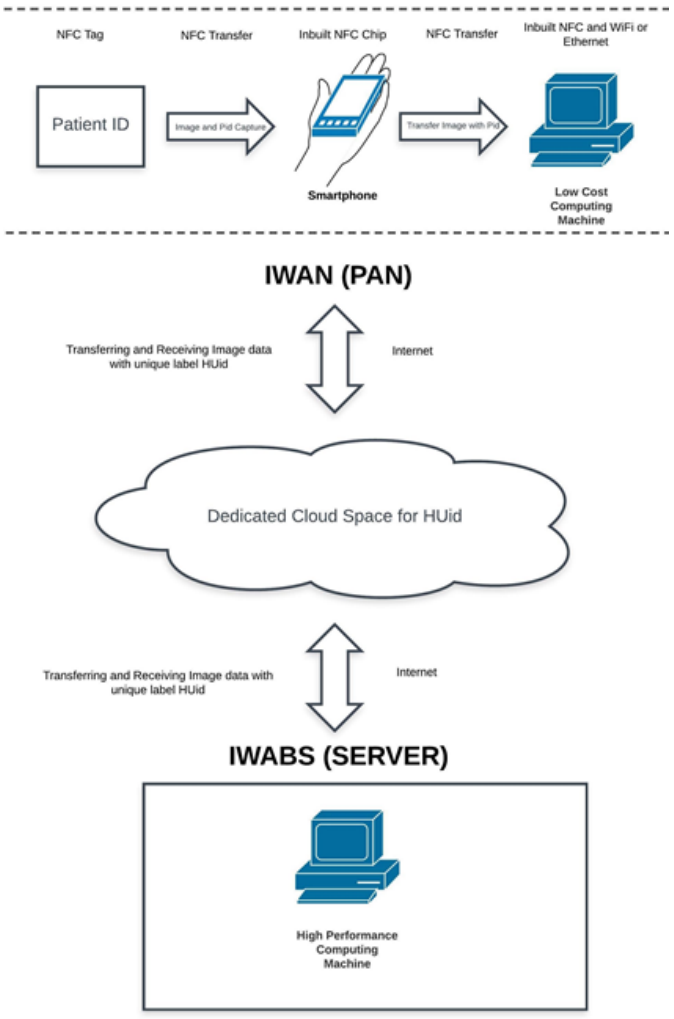


FIGURE 9.3: Comparison of Size of DFU against the size of image

area, ischemia, infection and depth of DFU. This diagnosis system could be deployed at the cloud server to remotely assess the DFU, provide faster feedback with good accuracy. The overview of this DFU diagnosis system is shown in Fig. 9.3. But, this integrated system should be tested and validated rigorously by podiatrists and medical experts, before it is implemented in the real healthcare setting and deployed as a mobile application.

Bibliography

[1] Lawrence A Lavery, David G Armstrong, and Lawrence B Harkless. Classification of diabetic foot wounds. *The Journal of Foot and Ankle Surgery*, 35 (6):528–531, 1996.

[2] Yann LeCun, Corinna Cortes, and Christopher JC Burges. The mnist database of handwritten digits, 1998.

[3] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[4] Brett Hewitt, Moi Hoon Yap, and Robyn Grant. Manual whisker annotator (mwa): A modular open-source tool. *Journal of Open Research Software*, 4 (1), 2016.

[5] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[6] Sarah Wild, Gojka Roglic, Anders Green, Richard Sicree, and Hilary King. Global prevalence of diabetes estimates for the year 2000 and projections for 2030. *Diabetes care*, 27(5):1047–1053, 2004.

[7] World Health Organization et al. Global report on diabetes who geneva, 2016.

[8] K Bakker, Jan Apelqvist, BA Lipsky, JJ Van Netten, and NC Schaper. The 2015 iwgdf guidance documents on prevention and management of foot problems in diabetes: development of an evidence-based global consensus. *Diabetes/metabolism research and reviews*, 32(S1):2–6, 2016.

[9] Andrew JM Boulton, Loretta Vileikyte, Gunnel Ragnarson-Tennvall, and Jan Apelqvist. The global burden of diabetic foot disease. *The Lancet*, 366 (9498):1719–1724, 2005.

[10] Florencia Aguiree, Alex Brown, Nam Ho Cho, Gisela Dahlquist, Sheree Dodd, Trisha Dunning, Michael Hirst, Christopher Hwang, Dianna Magliano, Chris Patterson, et al. *IDF Diabetes Atlas: sixth edition*. International Diabetes Federation, 6th edition, 2013.

[11] David G Armstrong, Andrew JM Boulton, and Sicco A Bus. Diabetic foot ulcers and their recurrence. *New England Journal of Medicine*, 376(24): 2367–2375, 2017.

[12] David G Armstrong, Lawrence A Lavery, and Lawrence B Harkless. Validation of a diabetic wound classification system: the contribution of depth, infection, and ischemia to risk of amputation. *Diabetes care*, 21(5):855–859, 1998.

[13] Peter Cavanagh, Christopher Attinger, Zulfiqarali Abbas, Arun Bal, Nina Rojas, and Zhang-Rong Xu. Cost of treating diabetic foot ulcers in five different countries. *Diabetes/metabolism research and reviews*, 28(S1):107–111, 2012.

[14] Chanjuan Liu, Jaap J van Netten, Jeff G Van Baal, Sicco A Bus, and Ferdi van Der Heijden. Automatic detection of diabetic foot complications with infrared thermography by asymmetric analysis. *Journal of biomedical optics*, 20(2):026003–026003, 2015.

[15] Lei Wang, Peder Pedersen, Emmanuel Agu, Diane Strong, and Bengisu Tulu. Area determination of diabetic foot ulcer images using a cascaded two-stage svm based classification. *IEEE Transactions on Biomedical Engineering*, 2016.

[16] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*, pages 818–833. Springer, 2014.

[17] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[18] Marios Anthimopoulos, Stergios Christodoulidis, Lukas Ebner, Andreas Christe, and Stavroula Mougiakakou. Lung pattern classification for interstitial lung diseases using a deep convolutional neural network. *IEEE transactions on medical imaging*, 35(5):1207–1216, 2016.

[19] Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5):1285–1298, 2016.

[20] Ezak Ahmad, Manu Goyal, Jamie S McPhee, Hans Degens, and Moi Hoon Yap. Semantic segmentation of human thigh quadriceps muscle in magnetic resonance images. *arXiv preprint arXiv:1801.00415*, 2018.

[21] Paras Lakhani and Baskaran Sundaram. Deep learning at chest radiography: automated classification of pulmonary tuberculosis by using convolutional neural networks. *Radiology*, 284(2):574–582, 2017.

[22] Moi Hoon Yap, Manu Goyal, Fatima Osman, Ezak Ahmad, Robert Martí, Erika Denton, Arne Juette, and Reyer Zwiggelaar. End-to-end breast ultrasound lesions recognition with a deep learning approach. In *Medical Imaging 2018: Biomedical Applications in Molecular, Structural, and Functional Imaging*, volume 10578, page 1057819. International Society for Optics and Photonics, 2018.

[23] Moi Hoon Yap, Manu Goyal, Fatima M Osman, Robert Martí, Erika Denton, Arne Juette, and Reyer Zwiggelaar. Breast ultrasound lesions recognition: end-to-end deep learning approaches. *Journal of Medical Imaging*, 6(1): 011007, 2018.

[24] Simon LF Walsh, Lucio Calandriello, Mario Silva, and Nicola Sverzellati. Deep learning for classifying fibrotic lung disease on high-resolution computed tomography: a case-cohort study. *The Lancet Respiratory Medicine*, 6(11):837–845, 2018.

[25] M. Goyal, M. H. Yap, N. D. Reeves, S. Rajbhandari, and J. Spragg. Fully convolutional networks for diabetic foot ulcer segmentation. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 618–623, Oct 2017. doi: 10.1109/SMC.2017.8122675.

[26] Manu Goyal, Neil D Reeves, Adrian K Davison, Satyan Rajbhandari, Jennifer Spragg, and Moi Hoon Yap. Dfunet: Convolutional neural networks for diabetic foot ulcer classification. *arXiv preprint arXiv:1711.10448*, 2017.

[27] Manu Goyal, Neil Reeves, Satyan Rajbhandari, and Moi Hoon Yap. Robust methods for real-time diabetic foot ulcer detection and localization on mobile devices. *IEEE journal of biomedical and health informatics*, 2018.

[28] F William Wagner. The diabetic foot. *Orthopedics*, 10(1):163–172, 1987.

[29] Robert G Frykberg. Diabetic foot ulcers: pathogenesis and management. *American family physician*, 66(9):1655–1662, 2002.

[30] Paul Ince, Zulfiqarali G Abbas, Janet K Lutale, Abdul Basit, Syed Mansoor Ali, Farooq Chohan, Stephan Morbach, Jörg Möllenberg, Fran L Game, and William J Jeffcoate. Use of the sinbad classification system and score in comparing outcome of foot ulcer management on three continents. *Diabetes care*, 31(5):964–967, 2008.

[31] Douglas A Perednia and Ace Allen. Telemedicine technology and clinical applications. *Jama*, 273(6):483–488, 1995.

[32] P Rubegni, N Nami, G Cevenini, S Poggiali, R Hofmann-Wellenhof, C Massone, R Bilenchi, M Bartalini, R Cappelli, and M Fimiani. Geriatric teledermatology: store-and-forward vs. face-to-face examination. *Journal of the European Academy of Dermatology and Venereology*, 25(11):1334–1339, 2011.

[33] David Moreno-Ramirez, Lara Ferrandiz, Adoracion Nieto-Garcia, Rafael Carrasco, Pedro Moreno-Alvarez, Rafael Galdeano, Esther Bidegain, Juan J Rios-Martin, and Francisco M Camacho. Store-and-forward teledermatology in skin cancer triage: experience and evaluation of 2009 teleconsultations. *Archives of Dermatology*, 143(4):479–483, 2007.

[34] Michael Shapiro, William D James, Rex Kessler, Francis C Lazorik, Kenneth A Katz, John Tam, David S Nieves, and Jeffrey J Miller. Comparison of skin biopsy triage decisions in 49 patients with pigmented lesions and skin neoplasms: store-and-forward teledermatology vs face-to-face dermatology. *Archives of dermatology*, 140(5):525–528, 2004.

[35] Jennifer L Hsiao and Dennis H Oh. The impact of store-and-forward teledermatology on skin cancer diagnosis and treatment. *Journal of the American Academy of Dermatology*, 59(2):260–267, 2008.

[36] Marilyn J Field and Jim Grigsby. Telemedicine and remote patient monitoring. *Jama*, 288(4):423–425, 2002.

[37] Vishal Nangalia, David R Prytherch, and Gary B Smith. Health technology assessment review: Remote monitoring of vital signs-current status and future challenges. *Critical Care*, 14(5):233, 2010.

[38] Catherine Klersy, Annalisa De Silvestri, Gabriella Gabutti, François Regoli, and Angelo Auricchio. A meta-analysis of remote monitoring of heart failure patients. *Journal of the American College of Cardiology*, 54(18):1683–1694, 2009.

[39] EJ Gomez, F Del Pozo, and ME Hernando. Telemedicine for diabetes care: the diabtel approach towards diabetes telecare. *Medical Informatics*, 21(4): 283–295, 1996.

[40] S Franc, A Daoudi, S Mounier, B Boucherie, H Laroye, C Peschard, D Dardari, O Juy, E Requeda, L Canipel, et al. Telemedicine: what more is needed for its integration in everyday life? *Diabetes & metabolism*, 37:S71–S77, 2011.

[41] Rolf Engelbrecht and Claudia Hildebrand. Telemedicine and diabetes. *Studies in health technology and informatics*, pages 142–154, 1999.

[42] Jane Clemensen, Simon B Larsen, Marit Kirkevold, and Niels Ejskjaer. Treatment of diabetic foot ulcers in the home: video consultations as an alternative to outpatient hospital care. *International journal of telemedicine and applications*, 2008:1, 2008.

[43] Frank L Bowling, Laurie King, James A Paterson, Jingyi Hu, Benjamin A Lipsky, David R Matthews, and Andrew JM Boulton. Remote assessment of diabetic foot ulcers using a novel wound imaging system. *Wound Repair and Regeneration*, 19(1):25–30, 2011.

[44] Constantijn EVB Hazenberg, Jaap J van Netten, Sjef G van Baal, and Sicco A Bus. Assessment of signs of foot infection in diabetes patients using

photographic foot imaging and infrared thermography. *Diabetes technology & therapeutics*, 16(6):370–377, 2014.

[45] Piotr Foltynski, Jan M Wojcicki, Piotr Ladyzynski, Karolina Migalska-Musial, Grzegorz Rosinski, Janusz Krzymien, and Waldemar Karnafel. Monitoring of diabetic foot syndrome treatment: some new perspectives. *Artificial organs*, 35(2):176–182, 2011.

[46] Jaap J van Netten, Damien Clark, Peter A Lazzarini, Monika Janda, and Lloyd F Reed. The validity and reliability of remote diabetic foot ulcer assessment using mobile phone images. *Scientific Reports*, 7(1):9480, 2017.

[47] Jaap J van Netten, Miranda Prijs, Jeff G van Baal, Chanjuan Liu, Ferdi van Der Heijden, and Sicco A Bus. Diagnostic values for skin temperature assessment to detect diabetes-related foot complications. *Diabetes technology & therapeutics*, 16(11):714–721, 2014.

[48] Michel H Hermans. Wounds and ulcers: back to the old nomenclature. *Wounds*, 22(11):289–93, 2010.

[49] Hazem Wannous, Yves Lucas, and Sylvie Treuillet. Enhanced assessment of the wound-healing process by accurate multiview tissue classification. *IEEE transactions on Medical Imaging*, 30(2):315–326, 2011.

[50] Marina Kolesnik and Ales Fexa. Multi-dimensional color histograms for segmentation of wounds in images. In *International Conference Image Analysis and Recognition*, pages 1014–1022. Springer, 2005.

[51] M. Kolesnik and A. Fexa. How robust is the svm wound segmentation? In *Signal Processing Symposium, 2006. NORSIG 2006. Proceedings of the 7th Nordic*, pages 50–53. IEEE, 2006.

[52] Elisabeth S Papazoglou, Leonid Zubkov, Xiang Mao, Michael Neidrauer, Nicolas Rannou, and Michael S Weingarten. Image analysis of chronic wounds for determining the surface area. *Wound repair and regeneration*, 18(4):349–358, 2010.

[53] Francisco Veredas, Héctor Mesa, and Laura Morente. Binary tissue classification on wound images with neural networks and bayesian classifiers. *IEEE transactions on medical imaging*, 29(2):410–427, 2010.

[54] Manoj Kumar Yadav, Dhane Dhiraj Manohar, Gargi Mukherjee, and Chandan Chakraborty. Segmentation of chronic wound areas by clustering techniques using selected color space. *Journal of Medical Imaging and Health Informatics*, 3(1):22–29, 2013.

[55] A Castro, Carmen Bóveda, and B Arcay. Analysis of fuzzy clustering algorithms for the segmentation of burn wounds photographs. In *International Conference Image Analysis and Recognition*, pages 491–501. Springer, 2006.

[56] Do Hyun Chung and Guillermo Sapiro. Segmenting skin lesions with partial-differential-equations-based image processing algorithms. *IEEE transactions on Medical Imaging*, 19(7):763–767, 2000.

[57] Changhan Wang, Xinchen Yan, Max Smith, Kanika Kochhar, Marcie Rubin, Stephen M Warren, James Wrobel, and Honglak Lee. A unified framework for automatic wound segmentation and analysis with deep convolutional neural networks. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, pages 2415–2418. IEEE, 2015.

[58] Manu Goyal and Moi Hoon Yap. Region of interest detection in dermoscopic images for natural data-augmentation. *arXiv preprint arXiv:1807.10711*, 2018.

[59] Jaap J van Netten, Jeff G van Baal, Chanjuan Liu, Ferdi van Der Heijden, and Sicco A Bus. Infrared thermal imaging for automated detection of diabetic foot complications, 2013.

[60] Chanjuan Liu, Ferdi van der Heijden, Marvin E Klein, Jeff G van Baal, Sicco A Bus, and Jaap J van Netten. Infrared dermal thermography on diabetic feet soles to predict ulcerations: a case study. In *Advanced Biomedical and Clinical Diagnostic Systems XI*, volume 8572, page 85720N. International Society for Optics and Photonics, 2013.

[61] JR Harding, DF Wertheim, RJ Williams, JM Melhuish, D Banerjee, and KG Harding. Infrared imaging in diabetic foot ulceration. In *Engineering in Medicine and Biology Society, 1998. Proceedings of the 20th Annual International Conference of the IEEE*, volume 2, pages 916–918. IEEE, 1998.

[62] D Hernandez-Contreras, H Peregrina-Barreto, J Rangel-Magdaleno, and J Gonzalez-Bernal. Narrative review: Diabetic foot and infrared thermography. *Infrared Physics & Technology*, 78:105–117, 2016.

[63] Muhammad Adam, Eddie YK Ng, Jen Hong Tan, Marabelle L Heng, Jasper WK Tong, and U Rajendra Acharya. Computer aided diagnosis of diabetic foot using infrared thermography: A review. *Computers in biology and medicine*, 91:326–336, 2017.

[64] Moi Hoon Yap, Choon-Ching Ng, Katie Chatwin, Caroline A Abbott, Frank L Bowling, Andrew JM Boulton, and Neil D Reeves. Computer vision algorithms in the detection of diabetic foot ulceration a new paradigm for diabetic foot care? *Journal of diabetes science and technology*, page 1932296815611425, 2015.

[65] Moi Hoon Yap, Katie E Chatwin, Choon-Ching Ng, Caroline A Abbott, Frank L Bowling, Satyan Rajbhandari, Andrew JM Boulton, and Neil D Reeves. footsnap: A new mobile application for standardizing diabetic foot images. *Journal of Diabetes Science and Technology*, page 1932296817713761, 2017.

[66] Ross Brown, Bernd Ploderer, Leonard Si Da Seng, Jaap J van Netten, and Peter A Lazzarini. Myfootcare: A mobile self-tracking tool to promote self-care amongst people with diabetic foot ulcers. 2017.

[67] John R Jensen and Kalmesh Lulla. Introductory digital image processing: a remote sensing perspective. 1987.

[68] Jae S Lim. Two-dimensional signal and image processing. *Englewood Cliffs, NJ, Prentice Hall, 1990, 710 p.*, 1990.

[69] Gregory A Baxes. *Digital image processing: principles and applications*. Wiley New York, 1994.

[70] Milan Sonka, Vaclav Hlavac, and Roger Boyle. *Image processing, analysis, and machine vision*. Cengage Learning, 2014.

[71] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[72] Nasser M Nasrabadi. Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4):049901, 2007.

[73] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[74] Rich Caruana and Alexandru Niculescu-Mizil. An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, pages 161–168. ACM, 2006.

[75] Gustavo Carneiro, Antoni B Chan, Pedro J Moreno, and Nuno Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 29(3):394–410, 2007.

[76] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. Semi-supervised learning (chapelle, o. et al., eds.; 2006)[book reviews]. *IEEE Transactions on Neural Networks*, 20(3):542–542, 2009.

[77] Robert M Haralick, Karthikeyan Shanmugam, et al. Textural features for image classification. *IEEE Transactions on systems, man, and cybernetics*, (6):610–621, 1973.

[78] Manu Goyal, Neil D Reeves, Adrian K Davison, Satyan Rajbhandari, Jennifer Spragg, and Moi Hoon Yap. Dfunet: Convolutional neural networks for diabetic foot ulcer classification. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2018.

[79] Manu Goyal and Moi Hoon Yap. Multi-class semantic segmentation of skin lesions via fully convolutional networks. *arXiv preprint arXiv:1711.10449*, 2017.

[80] Yun Fu and Thomas S Huang. Human age estimation with regression on discriminative aging manifold. *IEEE Transactions on Multimedia*, 10(4): 578–584, 2008.

[81] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. Unsupervised learning. In *The elements of statistical learning*, pages 485–585. Springer, 2009.

[82] Quoc V Le. Building high-level features using large scale unsupervised learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8595–8598. IEEE, 2013.

[83] HP Ng, SH Ong, KWC Foong, PS Goh, and WL Nowinski. Medical image segmentation using k-means clustering and improved watershed algorithm. In *Image Analysis and Interpretation, 2006 IEEE Southwest Symposium on*, pages 61–65. IEEE, 2006.

[84] Anil K Jain. Data clustering: 50 years beyond k-means. *Pattern recognition letters*, 31(8):651–666, 2010.

[85] Kardi Teknomo. K-means clustering tutorial. *Medicine*, 100(4):3, 2006.

[86] Wolfgang Förstner. A framework for low level feature extraction. In *European Conference on Computer Vision*, pages 383–394. Springer, 1994.

[87] Zhenhua Guo, Lei Zhang, and David Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, 2010.

[88] Judson P Jones and Larry A Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6):1233–1258, 1987.

[89] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[90] Dana H Ballard. Generalizing the hough transform to detect arbitrary shapes. *Pattern recognition*, 13(2):111–122, 1981.

[91] Yu-Ichi Ohta, Takeo Kanade, and Toshiyuki Sakai. Color information for region segmentation. *Computer graphics and image processing*, 13(3):222–241, 1980.

[92] Dong-Chen He and Li Wang. Texture unit, texture spectrum, and texture analysis. *IEEE transactions on Geoscience and Remote Sensing*, 28(4):509–512, 1990.

[93] Caifeng Shan, Shaogang Gong, and Peter W. McOwan. Facial expression recognition based on Local Binary Patterns: A comprehensive study. *Image*

and Vision Computing, 27(6):803–816, 2009. ISSN 02628856. doi: 10.1016/j.imavis.2008.08.005. URL http://dx.doi.org/10.1016/j.imavis.2008.08.005.

[94] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE, 2005.

[95] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (12):2037–2041, 2006.

[96] Matti Pietikäinen, Abdenour Hadid, Guoying Zhao, and Timo Ahonen. *Computer vision using local binary patterns*, volume 40. Springer Science & Business Media, 2011.

[97] Marko Heikkilä, Matti Pietikäinen, and Cordelia Schmid. Description of interest regions with local binary patterns. *Pattern recognition*, 42(3):425–436, 2009.

[98] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[99] Lipo Wang. *Support vector machines: theory and applications*, volume 177. Springer Science & Business Media, 2005.

[100] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[101] Erico Guizzo. How googles self-driving car works. *IEEE Spectrum Online*, 18(7):1132–1141, 2011.

[102] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Brian Kingsbury, et al. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal processing magazine*, 29, 2012.

[103] Francois Chollet. Building powerful image classification models using very little data. *The Keras Blog, 05/06/2015*, 2016.

[104] Kai Zhao, Denis Khryashchev, Juliana Freire, Claudio Silva, and Huy Vo. Predicting taxi demand at high spatial resolution: Approaching the limit

of predictability. In *2016 IEEE International Conference on Big Data (Big Data)*, pages 833–842. IEEE, 2016.

[105] Stuart J Russell and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[106] Vincent Weidlich and Georg A Weidlich. Artificial intelligence in medicine and radiation oncology. *Cureus*, 10(4), 2018.

[107] Varun Gulshan, Lily Peng, Marc Coram, Martin C Stumpe, Derek Wu, Arunachalam Narayanaswamy, Subhashini Venugopalan, Kasumi Widner, Tom Madams, Jorge Cuadros, et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *Jama*, 316(22):2402–2410, 2016.

[108] Christian Herweh, Peter A Ringleb, Geraldine Rauch, Steven Gerry, Lars Behrens, Markus Möhlenbruch, Rebecca Gottorf, Daniel Richter, Simon Schieber, and Simon Nagel. Performance of e-aspects software in comparison to that of stroke physicians on assessing ct scans of acute ischemic stroke patients. *International Journal of Stroke*, 11(4):438–445, 2016.

[109] Michael Chappell, Mark Woolrich, and Thomas Okell. Fast analysis method for non-invasive imaging of blood flow using vessel-encoded arterial spin labelling, September 12 2017. US Patent 9,757,047.

[110] Monique Thissen, Andreea Udrea, Michelle Hacking, Tanja von Braunmuehl, and Thomas Ruzicka. mhealth app for risk assessment of pigmented and nonpigmented skin lesions a study on sensitivity and specificity in detecting malignancy. *Telemedicine and e-Health*, 23(12):948–954, 2017.

[111] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[112] Annegreet Van Opbroek, M Arfan Ikram, Meike W Vernooij, and Marleen De Bruijne. Transfer learning improves supervised image segmentation across imaging protocols. *IEEE transactions on medical imaging*, 34(5): 1018–1030, 2015.

[113] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[114] John C Platt. 12 fast training of support vector machines using sequential minimal optimization. *Advances in kernel methods*, pages 185–208, 1999.

[115] Yann LeCun, Yoshua Bengio, et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[116] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014.

[117] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[118] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[119] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2), 2012.

[120] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[121] Forrest Iandola, Matt Moskewicz, Sergey Karayev, Ross Girshick, Trevor Darrell, and Kurt Keutzer. Densenet: Implementing efficient convnet descriptor pyramids. *arXiv preprint arXiv:1404.1869*, 2014.

[122] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang. Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE transactions on medical imaging*, 35(5):1299–1312, 2016.

[123] David L Steed, Dennis Donohoe, Marshall W Webster, and Linda Lindsley. Effect of extensive debridement and treatment on the healing of diabetic foot ulcers. diabetic ulcer study group. *Journal of the American College of Surgeons*, 183(1):61–64, 1996.

[124] SM Rajbhandari, ND Harris, M Sutton, C Lockett, S Eaton, M Gadour, S Tesfaye, and JD Ward. Digital imaging: an accurate and easy method of measuring foot ulcers. *Diabetic medicine*, 16(4):339–342, 1999.

[125] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.

[126] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[127] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[128] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[129] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[130] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.

[131] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[132] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[133] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[134] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, pages 4278–4284, 2017.

[135] Jonathan Huang, Vivek Rathod, Chen Sun, Menglong Zhu, Anoop Korattikara, Alireza Fathi, Ian Fischer, Zbigniew Wojna, Yang Song, Sergio Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. *arXiv preprint arXiv:1611.10012*, 2016.

[136] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.

[137] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.

[138] Remco R Bouckaert. Bayesian network classifiers in weka. 2004.

[139] Dennis W Ruck, Steven K Rogers, Matthew Kabrisky, Mark E Oxley, and Bruce W Suter. The multilayer perceptron as an approximation to a bayes optimal discriminant function. *IEEE Transactions on Neural Networks*, 1 (4):296–298, 1990.

[140] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[141] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The weka data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18, 2009.

[142] Rohit Arora. Comparative analysis of classification algorithms on different datasets using weka. *International Journal of Computer Applications*, 54 (13), 2012.

[143] G Sahoo and Yugal Kumar. Analysis of parametric & non parametric classifiers for classification technique using weka. *International Journal of Information Technology and Computer Science (IJITCS)*, 4(7):43, 2012.

[144] Christian Szegedy, Sergey Ioffe, and Vincent Vanhoucke. Inception-v4, inception-resnet and the impact of residual connections on learning. *CoRR*, abs/1602.07261, 2016. URL http://arxiv.org/abs/1602.07261.